

Handleiding Möbius Deel C

**Copyright © Metha Kamminga
nov. 2022**

Handleiding Möbius Deel C

Contents

1 Möbius Toets items maken deel C	1
1.1 Dynamische figuren	1
1.1.1 Grafieken met Maple	1
1.1.1.1 Tips voor grafieken met Maple	1
1.1.1.2 Eenvoudige grafiek van een functie	2
1.1.1.3 Gridlines	3
1.1.1.4 Meer grafieken in één figuur	4
1.1.1.5 Grafieken van impliciete functies	6
1.1.1.6 Ruimtefiguur	8
1.1.1.6.1 Impliciete functie en een functie van twee variabelen	8
1.1.1.6.2 Ruimtefiguur cilinder en plat vlak met verschillende oriëntatie	9
1.1.1.6.3 Ruimtefiguur met snijkromme van parabolische cilinder en gebogen vlak	11
1.1.1.7 Grafieken met asymptoten	12
1.1.1.8 Gekleurde gebieden in de grafiek	13
1.1.1.8.1 Enkelvoudig gebied	13
1.1.1.8.2 Meervoudig gebied	14
1.1.1.8.3 Werken met transparantie	15
1.1.1.8.4 Gebieden afdekken met wit	17
1.1.1.8.5 Gebieden tussen krommen van impliciete functies	18
1.1.1.8.6 Gebieden kleuren bij (lineaire) ongelijkheden	19
1.1.1.9 Grafieken van stuksgewijze functies	20
1.1.1.9.1 De stuksgewijze functie	20
1.1.1.9.2 Met behulp van de stapfunctie (Heaviside)	21
1.1.1.9.3 Absolute waarde-functie	22
1.1.1.10 Logaritmische schalen	23
1.1.1.10.1 Enkellogaritmische schaal	23
1.1.1.10.2 Dubbellogaritmische schaal	24
1.1.1.11 Plottools	25
1.1.1.11.1 Polygoon	25
1.1.1.11.2 Driehoeken en tekst in een cirkel	26
1.1.1.11.3 Sector	26
1.1.1.11.4 Cirkel, vierkant en lijnstuk	27
1.1.1.12 Plottools 3D	28
1.1.1.12.1 Cilinder met plottools	28
1.1.1.12.2 Kegel met plottools	29
1.1.1.13 Plottools vectoren	31
1.1.1.14 Geometrie	32
1.1.1.15 Animaties	33
1.1.1.16 Lijnelementenveld	34
1.1.1.17 Complexplot	35
1.1.1.18 Eigenplot	37
1.1.2 Labeled Images	39
1.1.3 Free Body Diagram	41
1.1.3.1 Werking van de Free Body Diagram-vraag	42
1.1.3.2 Het ontwerpen van een Free Body Diagram-vraag	43
1.1.4 Sketch	47
1.1.4.1 Grafieken van functies schetsen	47
1.1.4.1.1 Ontwerp van de vraag om een functie te schetsen	49
1.1.4.1.2 Meer grafieken in één figuur	50
1.1.4.2 Stuksgewijze grafieken schetsen	51

1.1.4.2.1 Stuksgewijze grafieken ontwerpen	52
1.1.4.3 Lineaire ongelijkheden gebieden schetsen	52
1.1.4.4 Achtergrondfunctie instellen	53
1.1.4.5 Aanpassingen in de broncode	54
1.1.4.5.1 Tolerantie aanpassentext	54
1.1.4.5.2 Aantal zichtbare knoppentext	54
1.1.5 Organische chemie structuurformules	55
1.1.6 MathApp-questions	56
1.1.6.1 Voorbeeld van een eenvoudig Maple-Worksheet t.b.v de Math App	59
1.1.6.2 Voorbeeld van een geavanceerd Math App Maple-Worksheet	61
1.1.6.2.1 Startup-code van de Math App	63
1.1.6.2.2 Initialiseren van parameters	66
1.1.7 SVG-figuren programmeren	68
1.1.7.1 Berekeningen aan een profiel met dynamische figuur	69
1.2 Werken met Maple Repositories	70
1.2.1 De module met procedures in Maple voorbereiden	71
1.2.2 Bestanden uploaden in de file manager	74
1.2.3 Gebruik van de bestanden in het algorithm	74
1.3 Intervallen en ongelijkheden met Maple-repository*	75
1.4 Tweetalige vraag	79
1.5 Kansrekenen en statistiek	79
1.5.1 Normale verdeling	80
1.5.2 Binomiale verdeling	82
1.5.3 Poisson verdeling	83
1.5.4 Students_t-verdeling	84
1.5.5 chi ² -verdeling	85
1.5.6 Data	85
1.5.7 Visualisaties	87
1.6 Statistiek	88
1.6.1 Beschrijvende statistiek	88
1.6.1.1 Datasets genereren	89
1.6.1.2 Staafdiagram	91
1.6.1.3 Staafdiagram als histogram en cumulatief diagram	92
1.6.1.4 Histogram van een dataset	94
1.6.1.5 Boxdiagram	95
1.6.1.6 Scatterplot en trendlijn	96
1.6.1.7 Centrummaten	98
1.6.1.8 Spreidingsmaten van een steekproef	100
1.6.2 Kansverdelingen	100
1.6.2.1 Standaardnormale verdeling	102
1.6.2.2 Normale verdeling algemeen	105
Index	107

List of Figures

Figure 1.1: Grafiek van een derdegraads functie met drie nulpunten	2
Figure 1.2: Figuur met gridlines 2	4
Figure 1.3: Meer grafieken in één figuur	5
Figure 1.4: Grafiek van impliciete functie	7
Figure 1.5: Ruimtefiguur	8
Figure 1.6: Cilinder en plat vlak doorsnede	10
Figure 1.7: Snijkromme van parabolische cilinder en gebogen vlak	11
Figure 1.8: Verticale asymptoot	13
Figure 1.9: Eenvoudig gebied tussen functies kleuren	14
Figure 1.10: Meervoudig gebied tussen functies kleuren	15
Figure 1.11: Gebieden kleuren met transparantie	16
Figure 1.12: Algoritme van gebieden kleuren met transparantie	17
Figure 1.13: Oppervlakte afdekken met wit	18
Figure 1.14: Kleuring van gebieden met impliciete functies	19
Figure 1.15: Kleuren van gebieden bij lineaire ongelijkheden	19
Figure 1.16: Stuksgewijze functie	21
Figure 1.17: Stuksgewijze functie met de stapfunctie	22
Figure 1.18: Absolute waarde-functie	22
Figure 1.19: Omzetten naar een stuksgewijze functie	23
Figure 1.20: Enkellogaritmische schaal	23
Figure 1.21: Dubbellogaritmische schaal	24
Figure 1.22: Figuur met polygoon en gridlines	25
Figure 1.23: Ingeschreven driehoeken	26
Figure 1.24: Gekleurd gebied met sectoren	27
Figure 1.25: Cirkel, vierkant en lijnstuk	28
Figure 1.26: Cilinder met plottools	28
Figure 1.27: kegel met plottools	29
Figure 1.28: Een cilinder en een vlak samen	30
Figure 1.29: Vectoren met plottools	31
Figure 1.30: title of the figure	32
Figure 1.31: Animaties met een animatiegif in Maple gemaakt	33
Figure 1.32: Lijnelementenveld	35
Figure 1.33: Complexe getallen in het complexe vlak	36
Figure 1.34: Eigenvectoren plot	38
Figure 1.35: Situatieschets zonder gegevens	39
Figure 1.36: Situatieschets met variabele gegevens	39
Figure 1.37: Algorithm van de belaste balk-reactiekracht	40
Figure 1.38: Pijlen in een Free Body Diagram	42
Figure 1.39: Ligger met box meer control points	43
Figure 1.40: Editen van de Free Body Diagram Response Area 1	44
Figure 1.41: Editen van de Free Body Diagram Response Area 2	45
Figure 1.42: Definiëren van krachten in Free Body Diagram	46
Figure 1.43: Editen van de Free Body Diagram Response Area 2	46
Figure 1.44: Sketchingvraag	47
Figure 1.45: Sketchingvraag paraboolsegment	48
Figure 1.46: Sketchingvraag ontwerpen	50
Figure 1.47: Sketchingvraag met stuksgewijze functies	51
Figure 1.48: Lineaire ongelijkheid met sketching	52
Figure 1.49: Achtergrondfunctie bij Sketching vraag	53
Figure 1.50: 3D structuurformule organische scheikunde	55
Figure 1.51: 2D structuurformule organische chemie	56

Figure 1.52: Math App in de vraag	57
Figure 1.53: Aanmaken van een Math App antwoordveld	58
Figure 1.54: Grading van een Math App vraag	59
Figure 1.55: Math App eenvoudig Maple-Worksheet	59
Figure 1.56: Eigenschappen van de componenten in het Maple Worksheet	60
Figure 1.57: Math App worksheet in Maple	61
Figure 1.58: Math App Plotcomponent	61
Figure 1.59: Knop in Maple voor de startup-code	63
Figure 1.60: Algorithmische variabelen in de Math App	67
Figure 1.61: SVG-figuur van een rechthoek	68
Figure 1.62: SVG-figuur	69
Figure 1.63: Navigeren naar het Excelbestand	71
Figure 1.64: Werkblad en Celbereik opgeven	72
Figure 1.65: Rechthoekig bereik	72
Figure 1.66: File manager met repository bestanden	74
Figure 1.67: Het .lib-bestand ophalen met de Designer van het Algorithm	75
Figure 1.68: Interval files in the File Manager	75
Figure 1.69: Maple grade interval	76
Figure 1.70: Algorithm	77
Figure 1.71: Maple graded inequality	78
Figure 1.72: Normale verdeling	81
Figure 1.73: De Students_t-verdeling	84
Figure 1.74: Permutatie van een lijst getallen	89
Figure 1.75: Data genereren willekeurig	90
Figure 1.76: Data genereren uit een verdeling	90
Figure 1.77: Staafdiagram	91
Figure 1.78: Staafdiagram als histogram	93
Figure 1.79: Histogram van dataset	95
Figure 1.80: Boxplot	96
Figure 1.81: Determinatiecoëfficiënt berekenen	97
Figure 1.82: Trendlijn	98
Figure 1.83: Cumulatieve kans bij standaardnormale verdeling	103
Figure 1.84: Inverse cumulatieve kans bij standaardnormale verdeling	104
Figure 1.85: Normale verdeling	105

1 Möbius Toets items maken deel C

november 20 22

1.1 Dynamische figuren

Er zijn vele mogelijkheden om met het wisselen van de algoritmische variabelen steeds bijpassende grafieken aan te bieden ten behoeve van het illustreren van de vraag of ter verduidelijking bij de feed back. Maple kan daarbij van dienst zijn, maar er zijn ook nog andere mogelijkheden.

Het is wel handig als u iets af weet van de mogelijkheden van randomiseren. Immers de kracht van dynamische plaatjes is dat de figuren zich steeds aanpassen aan de veranderende waarden van de variabelen. Niet alleen getallen maar ook tekst kan variëren.

Er is een aparte handleiding om alles te weten te komen over het randomiseren van items in Möbius.

1.1.1 Grafieken met Maple

Hebt u weinig verstand van zaken om met Maple grafieken te maken, kijk dan eens in het volgende boek waar hoofdstuk 3 geheel gewijd is aan het maken van grafieken.

Handleiding Maple 16, Metha Kamminga

ISBN: 9789039526750

Uitgeverij Boom

1.1.1.1 Tips voor grafieken met Maple

Allerlei grafieken zijn te maken met het computeralgebrasysteem Maple dat in Möbius te gebruiken is.

Grafieken ter illustratie van een wiskundig geïntereerd vraagstuk kunnen vaak gemakkelijk met behulp van de grafische mogelijkheden van Maple gemaakt worden. In het *Algorithm* kan een dergelijke grafiek voorbereid worden. Dit kan bij elk content item verwezenlijkt worden en overall in de vraag of in de feedback of in de hints kan de variabele aangeroepen worden met de naam van de variabele en verschijnt de grafiek.

U geeft in het *Algorithm* een maple opdracht voor een grafiek met het volgende commando:

```
$figuur1=plotmaple(".....");
```

Op de plaats van de stippen komt de Maple-opdracht voor het maken van de grafiek.

In de volgende secties maakt u kennis met verschillende soorten grafieken.

TIP: Er is één belangrijk aandachtspunt: waar u binnen het computeralgebrasysteem Maple zelf een opdracht geeft waarin dubbele quotes gebruikt worden, gebruikt u in Möbius enkele quotes (of een andere manier om tekst als string weg te schrijven.)

TIP: Er zijn plotcommando's in Maple die direct te gebruiken zijn. Maar er zijn ook plotcommando's die ingedeeld zijn in pakketten zoals het plot-pakket, het plottools-pakket en dergelijke.

Met bijvoorbeeld `with(plots):` (afgesloten met dubbele punt) kunt u alvast het plot pakket activeren waarna u bijvoorbeeld het commando `display` in dat pakket direct kunt gebruiken.

U kunt in plaats daarvan ook de lange vorm hanteren met `plots[display](...)` zonder eerst het plot-pakket apart te activeren om toch het commando `display` te kunnen gebruiken.

1.1.1.2 Eenvoudige grafiek van een functie

U hebt een functie van één variabele (x). Hiervan is gemakkelijk een grafiek te maken.

In de volgende opgave wordt gevraagd de formule van de functie in te tikken in het Maple-graded antwoordveld.

Als illustratie is de grafiek van de bedoelde functie alvast gegeven.

Er is ook nog de mogelijkheid voor de student om zijn antwoord te checken door een grafiek van zijn antwoord te zien in een pop-up-venster zoals in de figuur te zien is.

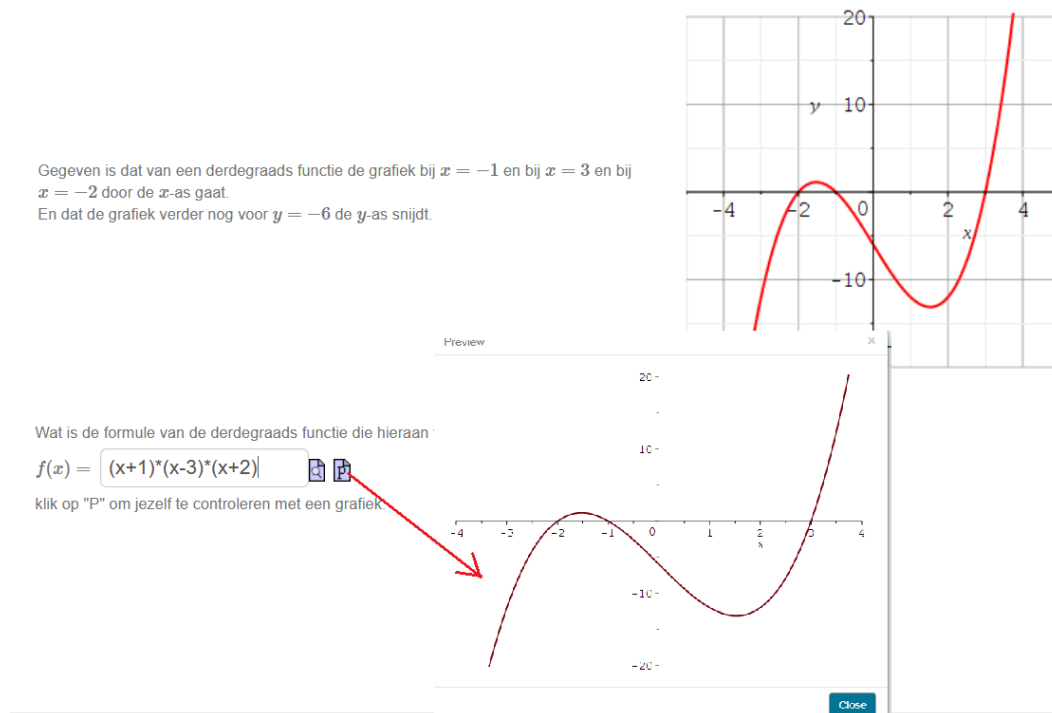


Figure 1.1: Grafiek van een derdegraads functie met drie nulpunten

Deze grafieken veranderen mee met de randomvariabelen.

In het *Algorithm* wordt de grafiek van de tekst als volgt voorbereid.

```
$lijstA="[-3, -2, -1, 1, 2, 3]";
$A=maple("StringTools[Randomize]():combinat[randperm]($lijstA)");
$a=switch(0, $A);
$b=switch(1, $A);
$c=switch(2, $A);
$d=switch(rint(2), range(-10, -1), range(1, 10));
$f=maple("-($d)/(($a)*($b)*($c))*(x-($a))*(x-($b))*(x-($c))");
$fdisplay=maple("printf(MathML:-ExportPresentation( $f))");
$xa=maple("printf(MathML:-ExportPresentation( x=$a))");
$xb=maple("printf(MathML:-ExportPresentation( x=$b))");
$xc=maple("printf(MathML:-ExportPresentation( x=$c))");
$yd=maple("printf(MathML:-ExportPresentation( y=$d))");
$kleur=switch(rint(3), "green", "red", "black");
$figuur=plotmaple("plot($f, x=-5..5, y=-20..20, color=$kleur,
thickness=2, gridlines=true),
```



```
plotoptions='height=300, width=300');
```

U maakt de grafiek met de naam \$figuur door binnen de opdracht `plotmaple(".....")` de maple-commando's te geven die nodig zijn voor het maken van de grafiek. Dat mogen ook meer opdrachten achter elkaar zijn. In dit voorbeeld is het een eenvoudige grafiek van de functie \$f.

Merk op dat de kleur hier zelfs gerandomiseerd is.

Als u alleen de commando's geeft voor het maken van de grafiek, dan wordt er een grafiek gegenereerd met een grootte van standaard ongeveer 500 pixels in het vierkant. Dat is echter vaak te groot. Binnen de opdracht `plotmaple("...")` kan nog een extra optie worden gegeven voor het formaat van de grafiek met de volgende toevoeging:

```
plotoptions='height=300, width=300'.
```

Deze optie komt buiten het Maple-commando `plot`, maar nog binnen de opdracht `plotmaple` te staan.

In de tekst van de vraag hoeft dan alleen nog de variabele \$figuur aangeroepen te worden.

Meestal biedt u die in een tabel aan met begeleidende tekst ernaast.

De grootte ervan is dus al op voorhand vastgelegd zie *Figure 1.1 (page 2)*

TIP: Bij dit antwoordveld (van het Maple-graded vraagtype) is het mogelijk om een pop-up aan te bieden, zodat de student de grafiek van zijn eigen ingevulde functie (\$RESPONSE) kan bekijken. Dat gaat met het invullen van de *Plotting Code* in het Maple-graded antwoordveld met de volgende Maple-opdracht:

```
plot(($RESPONSE), x=-4..4, -20..20, thickness=2);
```

(Als u niets invuld bij de *Plotting Code*, is de P voor het plotten van het ingevulde antwoord niet actief bij dit vraagtype.)

1.1.1.3 Gridlines

Het is gemakkelijk om gridlines in de figuur te krijgen door in het plotcommando de optie `gridlines=true` mee te geven.

Als u een specifieke kleur wilt hebben geeft u bijvoorbeeld de optie `[gridlines = [color = gray]]` mee.

Zie verder in de helpfunctie van het computeralgebrasysteem Maple voor eventueel nog meer variatie in gridlines.

Met het opgeven van `tickmarks` is het aantal gridlines te beïnvloeden zoals te zien is in de volgende figuur met daaronder het *Algorithm* met de variabelen om de figuur te genereren.

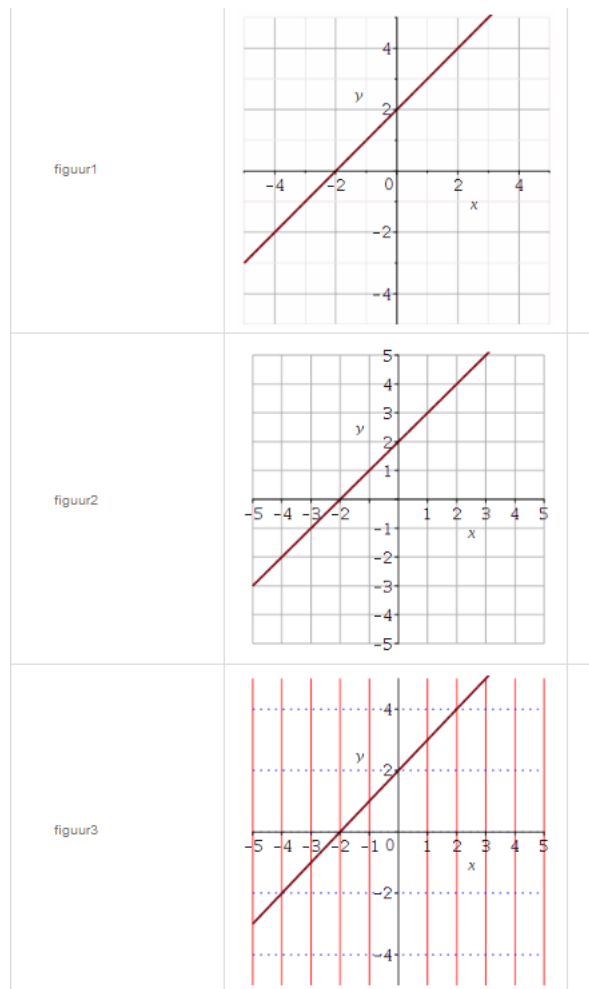


Figure 1.2: Figuur met gridlines 2

```
$a=switch(rint(2),range(-3,-1),range(1,3));
$b=range(-3,3);
$p=maple("($a)*x+($b)");
$figuur1=plotmaple("plot($p,x=-5..5,y=-5..5,thickness=2,gridlines=true),
plotoptions='height=300, width=300'");
$figuur2=plotmaple("plot($p,x=-5..5,y=-5..5,thickness=2,gridlines=true,
tickmarks=[10,10]),
plotoptions='height=300, width=300'");
$figuur3=plotmaple("plot($p,x=-5..5,y=-5..5,thickness=2,
axis[1] = [gridlines = [10,color=red]],
axis[2] = [gridlines = [ 5,linestyle=2,color=blue]]),
plotoptions='height=300, width=300'");
```

TIP: Zie ook in paragraaf (page 23) voor bijvoorbeeld het aanpassen van gridlines bij logaritmische schaal.

1.1.1.4 Meer grafieken in één figuur

In het volgende voorbeeld ziet u hoe u twee grafieken met verschillende kleur én nog een (groen) punt in één figuur krijgt.

In het *Algorithm* vult u het volgende in:

```
$Px=range(-8,8);
$Py=range(-8,8);
$Qx=switch(rint(2), range(-9,$Px-1), range($Px+1,9));
$Qy=range(-8,8);
$a=range(-3,3);
$rc1=$Py-($Qy);
$rc2=$Px-($Qx);
$rc=maple("($Py-($Qy))/($Px-($Qx))");
$vergvraag=maple("$rc2*y-($rc1)*x=$a");
$functievraag=maple("solve($vergvraag,y)");
$functieantwoord=maple("$rc*(x-($Px))+($Py)");
$plot=plotmaple("
p1:=plot([$functievraag,$functieantwoord], x=-8..8, -8..8,
color=[red, blue], legend=[typeset(lijn[1]),typeset(lijn[2])]):
p2:=plot({[$Px,$Py]}, style=point,color=green,symbolsize=20):
plots[display]({p1,p2},gridlines=true),
plotplotoptions='height=350, width=250'");
```

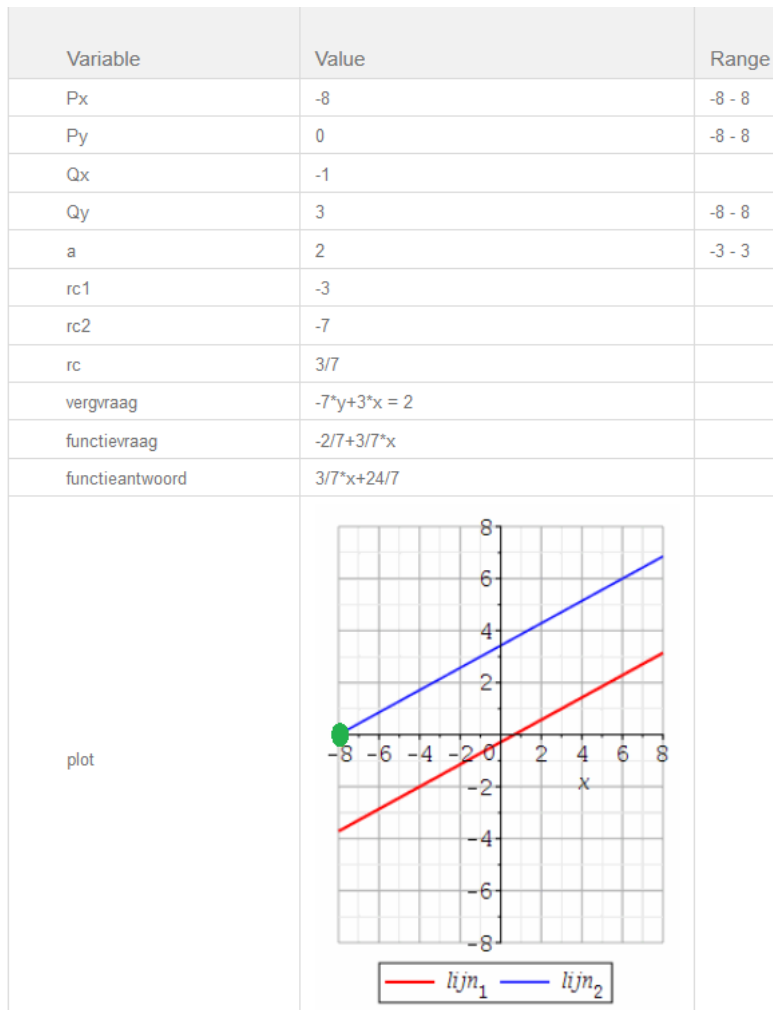


Figure 1.3: Meer grafieken in één figuur

In het *Algorithm* ziet u dat er **binnen Maple** een grafiek als variabele `p1` wordt gedeclareerd (met dubbele punt en `=`-teken en zonder `$`-teken).

In het plotcommando voor de lijnen zijn twee functies in een lijstje opgegeven:

```
[ $functievraag, $functieantwoord ] .
```

Het is dus mogelijk om meer functies te plotten in één plotcommando. De kleuren van de twee grafieken van deze twee functies worden ook met een lijstje opgegeven zodat alles in de pas loopt. Met de optie `legend` geeft u in een lijst de namen (of formules) van de functies.

Vervolgens declareert u **binnen Maple** nóg een grafiek als variabele `p2` en dat is een aparte grafiek met één punt in de kleur groen.

Deze twee verschillende grafieken 'legt u over elkaar heen' met behulp van het Maple-commando `display` uit het plot-pakket:

```
plots[display]({p1, p2}) en eventueel geeft u daarbij de optie gridlines=true.
```

TIP: Binnen een Maple-opdracht gebruikt u dus declaratie `p1:=...` (met dubbele punt en `=`-teken) en geen dollarteken voor het declareren van een (locale) variabele.

TIP: Een verzameling grafieken (met accolades) van verschillende soort legt u dus met `plots[display]({...})` 'over elkaar heen' (of met een lijst van grafieken (met rechte haken) `plots[display]( [...])`).

Dit kan ook in één plotcommando zonder tussentijdse declaraties, maar minder overzichtelijk op de volgende alternatieve manier:

```
$plot1=plotmaple("plots[display](
{plot([$functievraag, $functieantwoord], x=-8..8, -8..8,
color=[red, blue],
legend=[typeset(lijn[1]), typeset(lijn[2])]),
plot({[$Px, $Py]}, style=point, color=green, symbolsize=20)},
gridlines=true), plotplotoptions='height=350, width=250'");
```

1.1.1.5 Grafieken van impliciete functies

Voor impliciete functies (in vergelijkingsvorm) heeft Maple het commando `implicitplot` uit het plots-pakket.

Daarin is het ook weer mogelijk om twee of meer impliciete functies in een lijst mee te geven waarbij de lijst met kleuren, diktes en lijnstijlen in de pas loopt.

Programmeer in het *Algorithm* de volgende regels:

```
$a=range(1, 4);
$b=range(1, 4);
$c=range(2, 4);
$f=maple("x^2+y^2=1");
$fdisplay=maple("printf(MathML:-ExportPresentation($f))");
$g=maple("(x-($a))^2+((y-($b))/($c))^2=1");
$gdisplay=maple("printf(MathML:-ExportPresentation($g))");
$figuur=plotmaple("plots[implicitplot]([$f, $g],
x=-1..1+$a, y=-3..$c+$b, scaling=constrained,
gridrefine=3, color=[black, blue], thickness=[2, 1],
linestyle=[1, 2], gridlines=true),
plotoptions='height=250, width=250'");
```

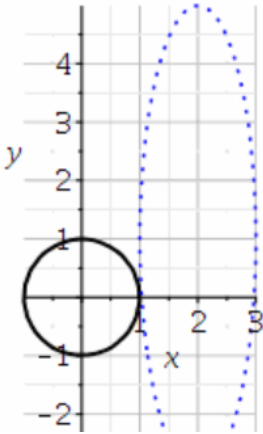
a	2	1 - 4
b	1	1 - 4
c	4	2 - 4
f	$x^2 + y^2 = 1$	
fdisplay	$x^2 + y^2 = 1$	
g	$(x-2)^2 + (1/4 y - 1/4)^2 = 1$	
gdisplay	$(x - 2)^2 + \left(\frac{1}{4} y - \frac{1}{4}\right)^2 = 1$	
figuur		

Figure 1.4: Grafiek van impliciete functie

Hier is met het Maple-commando `implicitplot` uit het plot-pakket gewerkt. In een lijst worden de twee vergelijkingen in x en y met behulp van de variabelen f en g opgegeven. Om goede grafieken te genereren geeft u het bereik voor x en y afhankelijk van de waarden van de randomvariabelen a en b en c .

TIP: Bij impliciete functies moet beslist altijd het bereik van de beide variabelen (x en y) opgegeven worden.

Om een echte cirkel ook in de figuur een echte cirkel te laten zijn is de optie `scaling=constrained` meegegeven. De assen zijn dan 1:1.

Voor de zekerheid is het aantal punten waarmee de grafieken zijn gemaakt iets opgehoogd met de optie `gridrefine=3` (standaard 2). Deze optie is speciaal voor impliciete functies en kunt u toepassen als de kromme niet geheel vloeiend verloopt, wat wijst op te weinig berekende punten. De lijst met kleuren loopt in de pas met de lijst van de functies. Let op dat de dikte en de stijl van de lijnen verschillend zijn.

Ook hier zijn buiten het plot-commando, maar nog binnen de opdracht `plotmaple("...")` de opties voor de grootte van de grafiek meegegeven (250×250 pixels).

TIP: Als u een Maple-graded vraagtype gebruikt om een vergelijking te graden, kan de student op *Plot* klikken om de grafiek te zien van zijn ingevoerde vergelijking.

De opdracht die bij *Plotting Code* dan gegeven moet worden is zuiver een Maple-commando en dat is hier:

```
plots[implicitplot]($RESPONSE, x=-1..$a+$c+1, y=-3..$b+1,
scaling=constrained, gridrefine=3, thickness=2);
```

Als u niets invult bij de *Plotting Code*, is de P achter het antwoordveld inactief.

TIP: Met het Maple-commando `plots[implicitplot3D]` uit het plotpakket is een vlak in de ruimte te tekenen met een vergelijking in 3 variabelen.

Zie voor meer informatie over voorbeelden met impliciete functies in paragraaf *Ruimtefiguur* (page 8).

1.1.1.6 Ruimtefiguur

In de volgende secties ziet u een paar voorbeelden over ruimtefiguren.

Voor 3D-grafieken die u met de muis kunt ronddraaien werkt u met een MathApp (zie paragraaf (page 56)).

1.1.1.6.1 Impliciete functie en een functie van twee variabelen

In het volgende voorbeeld ziet u het verband tussen een impliciete functie en een functie van twee variabelen.

Bij een constante waarde van de functie $f(x, y)$ krijgt u een bepaalde isolijn in het x, y -vlak met het impliciete functie plotcommando `implicitplot` uit het plot-pakket (zie paragraaf (page 6)).

Voor een aantal **contourlijnen** in het vlak is er een handig Maple-commando `contourplot` uit het plot-pakket.

De grafiek van de functie van twee variabelen is een 3D-figuur te visualiseren met het Maple-commando `plot3d`.

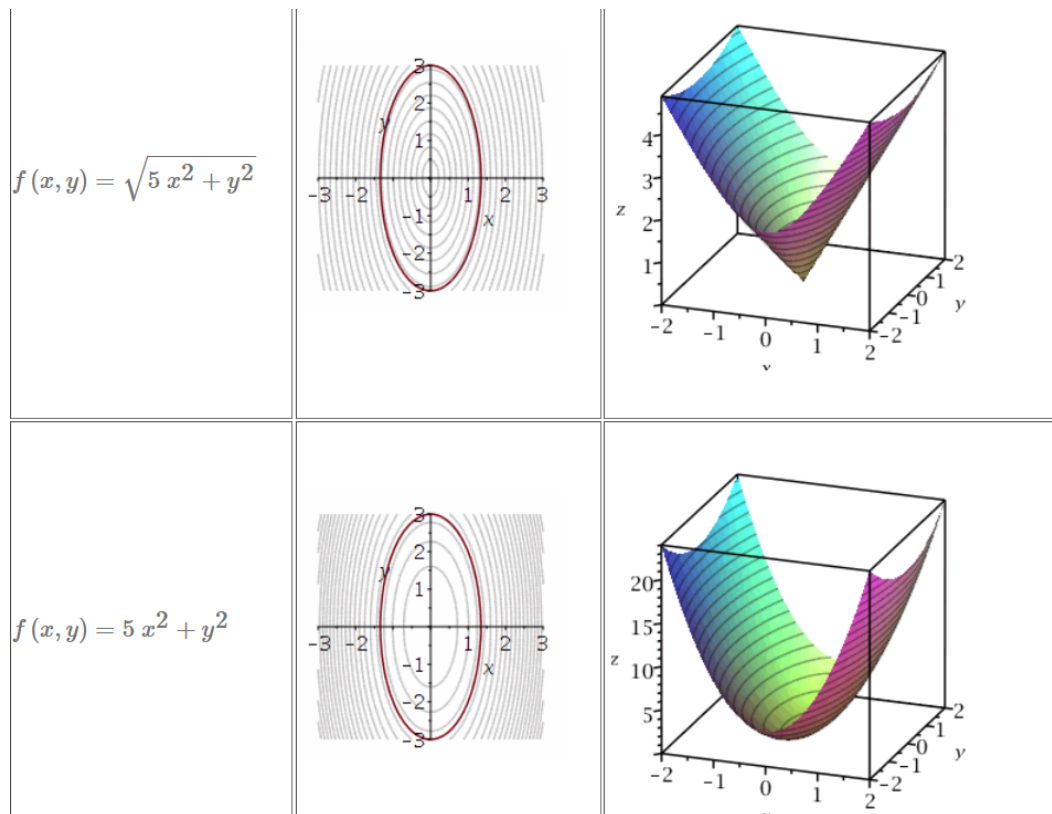


Figure 1.5: Ruimtefiguur

In het *Algorithm* programmeert u het volgende:

```
$lijstA="[0, 1, 2, 3]";
```

```

$A=maple("StringTools[Randomize]():combinat[randperm]($lijstA)");
$indexA1=switch(0,$A);
$indexA2=switch(1,$A);
$indexA3=switch(2,$A);
$indexA4=switch(3,$A);
$a=range(2,5);
$F1="x^2+$a*y^2";
$F2="$a*x^2+y^2";
$F3="sqrt(x^2+$a*y^2)";
$F4="sqrt($a*x^2+y^2)";
$f1=switch($indexA1,"$F1","$F2","$F3","$F4");
$f2=switch($indexA2,"$F1","$F2","$F3","$F4");
$W1=maple("simplify(eval($f1,{x=1,y=2}))");
$W2=maple("simplify(eval($f2,{x=1,y=2}))");
$f1display=maple("printf(MathML[ExportPresentation](f(x,y)=
$f1))");
$f2display=maple("printf(MathML[ExportPresentation](f(x,y)=
$f2))");
$p1=plotmaple("plots[display]([plots[contourplot]($f1,
x=-3..3,y=-3..3,color=grey, contours=20,scaling=constrained),
plots[implicitplot]($f1=$W1,x=-3..3,y=-3..3)]),
plotoptions='height=200, width=200'");
$P1=plotmaple("plot3d($f1,x=-2..2,y=-2..2,
orientation=[-70,70],lightmodel=light1,
style=patchcontour,contours=20,labels=[x,y,z]),
plotoptions='height=300, width=300'");
$p2=plotmaple("plots[display]([plots[contourplot]($f2,
x=-3..3,y=-3..3,color=grey, contours=20,scaling=constrained),
plots[implicitplot]($f2=$W2,x=-3..3,y=-3..3)]),
plotoptions='height=200, width=200'");
$P2=plotmaple("plot3d($f2,x=-2..2,y=-2..2,
orientation=[-70,70],lightmodel=light1,
style=patchcontour,contours=20,labels=[x,y,z]),
plotoptions='height=300, width=300'");

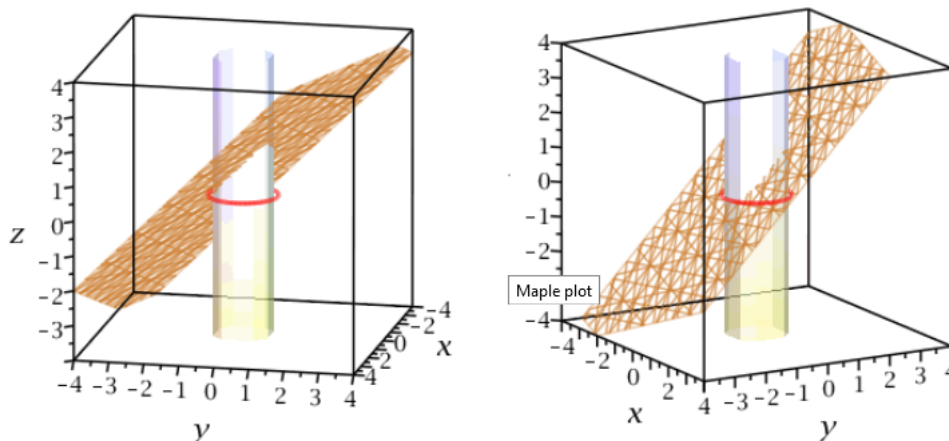
```

TIPS: Met de optie `style=patchcontour` voor de 3D-figuur krijgt u behalve het gekleurde oppervlak ook een aantal contourlijnen in het vlak van de 3D-figuur.
 Met de optie `orientation=[-70,70]` kunt u variëren en zien of de 3D-figuur goed zichtbaar is.
 Met de optie `lightmodel=light1` is de belichting en de kleurstelling van het 3D-vlak te beïnvloeden.

1.1.1.6.2 Ruimtefiguur cilinder en plat vlak met verschillende oriëntatie

Het wil ook wel eens helpen om twee of meer figuren aan te bieden met verschillende oriëntatie als het om 3D-figuren gaat, zoals in het volgende voorbeeld waarbij een cilinder doorsneden wordt met een vlak.

Preview



In de figuur zie je twee gezichtspunten van de situatie van een cilinder die doorsneden wordt met een vlak met vergelijking $x + 3y - 3z = -2$.

Het cirkelvormige 'grondvlak' van de cilinder ligt in het x, y -vlak.

Bereken de oppervlakte van de ellips waarmee het gegeven vlak de cilinder doorsnijdt.

Oppervlakte doorsnijdingsvlak is (geef het antwoord afgerond op 2 decimalen)

Figure 1.6: Cilinder en plat vlak doorsnede

In het *Algorithm* programmeert u de volgende regels om deze twee figuren (\$plot1 en \$plot2) te krijgen die hetzelfde zijn, maar met verschillende oriëntatie.

```
$n1=switch(rint(2), range(-3, -1), range(1, 3));
$n2=switch(rint(2), range(-3, -1), range(1, 3));
$n3=switch(rint(2), range(-3, -1), range(1, 3));
$n0=switch(rint(2), range(-3, -2), range(2, 3));
$vlak="($n1)*x+($n2)*y+($n3)*z=$n0";
$vlakdisplay=maple("printf(MathML[ExportPresentation]($vlak))");
$index=rint(3);
$spc=switch($index, "[cos(t), sin(t), 0]", "[cos(t), 0, sin(t)]", "[0, cos(t), sin(t)]");
$cil=switch($index, "x^2+y^2=1", "x^2+z^2=1", "z^2+y^2=1");
$grond=switch($index, "\ (x, y) -vlak", "\ (x, z) -vlak", "\ (y, z) -vlak");
$grvl=switch($index, "z=0", "y=0", "x=0");
$plot1=plotmaple("
p1:=plots[implicitplot3d]
($vlak, x=-4..4, y=-4..4, z=-4..4, color=gold,
style=wireframe, color=gold):
p2:=plots[implicitplot3d]($cil, x=-4..4, y=-4..4, z=-4..4,
lightmodel=light1, style=patchnogrid, transparency=0.6):
p3:=plots[spacecurve]($spc, t=0..2*Pi, color=red, thickness=3):
plots[display]([p1, p2, p3],
scaling=constrained, labels=[x, y, z],
labelfont=[Times, 15], orientation=[12, 76]),
plotoptions='height=350, width=350');
$plot2=plotmaple("
p1:=plots[implicitplot3d]
($vlak, x=-4..4, y=-4..4, z=-4..4, color=gold,
style=wireframe, color=gold):
p2:=plots[implicitplot3d]($cil, x=-4..4, y=-4..4, z=-4..4,
lightmodel=light1, style=patchnogrid, transparency=0.6):
p3:=plots[spacecurve]($spc, t=0..2*Pi, color=red, thickness=3):
plots[display]([p1, p2, p3],
scaling=constrained, labels=[y, z, x],
labelfont=[Times, 15], orientation=[76, 12]),
plotoptions='height=350, width=350');
```



```

p1:=plots[implicitplot3d]($vlak,x=-4..4,y=-4..4,z=-4..4,
color=gold,style=wireframe,color=gold):
p2:=plots[implicitplot3d]($cil,x=-4..4,y=-4..4,z=-4..4,
lightmodel=light1,style=patchnogrid,transparency=0.6):
p3:=plots[spacecurve]($spc,t=0..2*Pi,color=red,thickness=3):
plots[display]([p1,p2,p3],scaling=constrained,labels=[x,y,z],
labelfont=[Times,15],orientation=[-30,76]),
plotoptions='height=350, width=350');

```

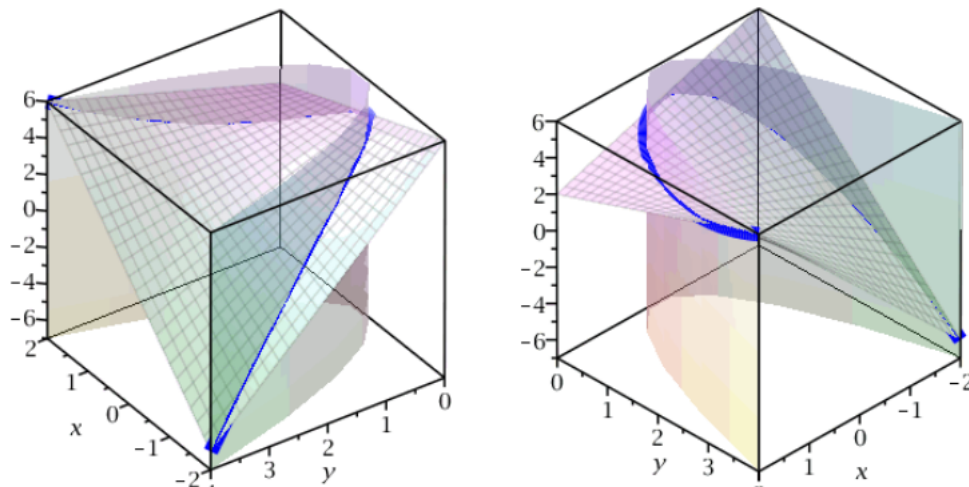
TIP: Bij het gebruik van het Maple-commando `implicitplot3d` uit het plotpakket moet voor alle drie variabelen het domein opgegeven worden.

1.1.1.6.3 Ruimtefiguur met snijkromme van parabolische cilinder en gebogen vlak

In het volgende voorbeeld wordt de snijkromme van een parabolische cilinder en een gebogen vlak gevraagd.

De figuur wordt duidelijker te interpreteren als u deze aanbiedt met twee oriëntaties.

Preview



In de figuur zie je twee gezichtspunten van de situatie van een parabolische cilinder $y = x^2$ doorsneden met het vlak $xy - x - y - z + 4 = 0$.

Geef de parametervoorstelling van de doorsnijdingskromme.

Stel daarbij $x = t$.

$x =$  
 $y =$  
 $z =$  

Figure 1.7: Snijkromme van parabolische cilinder en gebogen vlak

Voor het maken van een dergelijke figuur met daarin de ruimtekromme in het *Algorithm* hebt u twee dezelfde plots nodig met verschillende oriëntatie.

De ruimtekromme is in feite een **parameterkromme** die u maakt met het Maple-commando `spacecurve` uit het plot-pakket.

In het Algorithm programmeert u als volgt:

```
$index=rint(2);
```

```

$b=switch($index,1,2);
$bk=$b^2;
$a=range(-5,5);
$index1=rint(2);
$f=switch($index1,"1+x*y-x-y+($a)","1-x*y+x-y+($a)");
$fdisplay=maple("printf(MathML[ExportPresentation]($f-z=0))");
$g="y=x^2";
$gdisplay=maple("printf(MathML[ExportPresentation]($g))");
$stel="(x=t)";
$x="t";
$y="t^2";
$z=maple("subs({x=t,y=t^2},$f)");
$xdisplay=maple("printf(MathML[ExportPresentation](x=$x))");
$ydisplay=maple("printf(MathML[ExportPresentation](y=$y))");
$zdisplay=maple("printf(MathML[ExportPresentation](z=$z))");
$spcrv=switch($index1,"[x,x^2,1+x^3-x-x^2+($a)]","[x,x^2,1-x^3+x-x^2+($a)]");
$plot1=plotmaple("
p1:=plot3d($f,x=-$b..$b,y=0..$b^2,transparency=0.7,style=patch):
p2:=plots[implicitplot3d]($g,x=-$b..$b,y=0..$b^2,z=-10+($a)..3+($a),transparency=0.7,style=patchnogrid):
p3:=plots[spacecurve]($spcrv,x=-$b..$b,thickness=10,color=blue):
plots[display]([p1,p2,p3],orientation=[145,56]),
plotoptions='height=350, width=350');
$plot2=plotmaple("
p1:=plot3d($f,x=-$b..$b,y=0..$b^2,transparency=0.7,style=patch):
p2:=plots[implicitplot3d]($g,x=-$b..$b,y=0..$b^2,z=-10+($a)..3+($a),transparency=0.7,style=patchnogrid):
p3:=plots[spacecurve]($spcrv,x=-$b..$b,thickness=10,color=blue):
plots[display]([p1,p2,p3],orientation=[45,56]),
plotoptions='height=350, width=350');

```

1.1.1.7 Grafieken met asymptoten

Asymptoten van een functie behoren in feite niet tot de grafiek van de functie.

Als u de asymptoten gestippeld wilt weergeven, maakt u daarvan een aparte grafiek en vervolgens legt u de lijst met verschillende grafieken 'over elkaar heen' met het Maple-commando `display` uit het plot-pakket.

Voor het plotten van de functie zonder dat de asymptoot zichtbaar is, geeft u de optie `display=true` mee zodat niet automatisch de berekende punten verbonden worden.

In het *Algorithm* programmeert u het volgende:

```

$a=range(2,8);
$b=range(2,8);
$c=switch(rint(2),range(2,8),range(-8,-2));
$f1=maple("$a-($b*x)/(x-($c))");
$f2=maple("($a-$b*x)/(x-($c))");
$f=switch(rint(2),"$f1","$f2");
$ha=maple("limit($f,x=infinity)");
$plot=plotmaple("plots[display]([
plot($f,x=$c-40..$c+40,-20..20,thickness=2,discont=true),
plot([$c,t,t=-20..20],linestyle=2,color=blue),
plot($ha,x=$c-40..$c+40,color=blue,linestyle=2)]),

```

```
plotoptions='height=350, width=350');
```

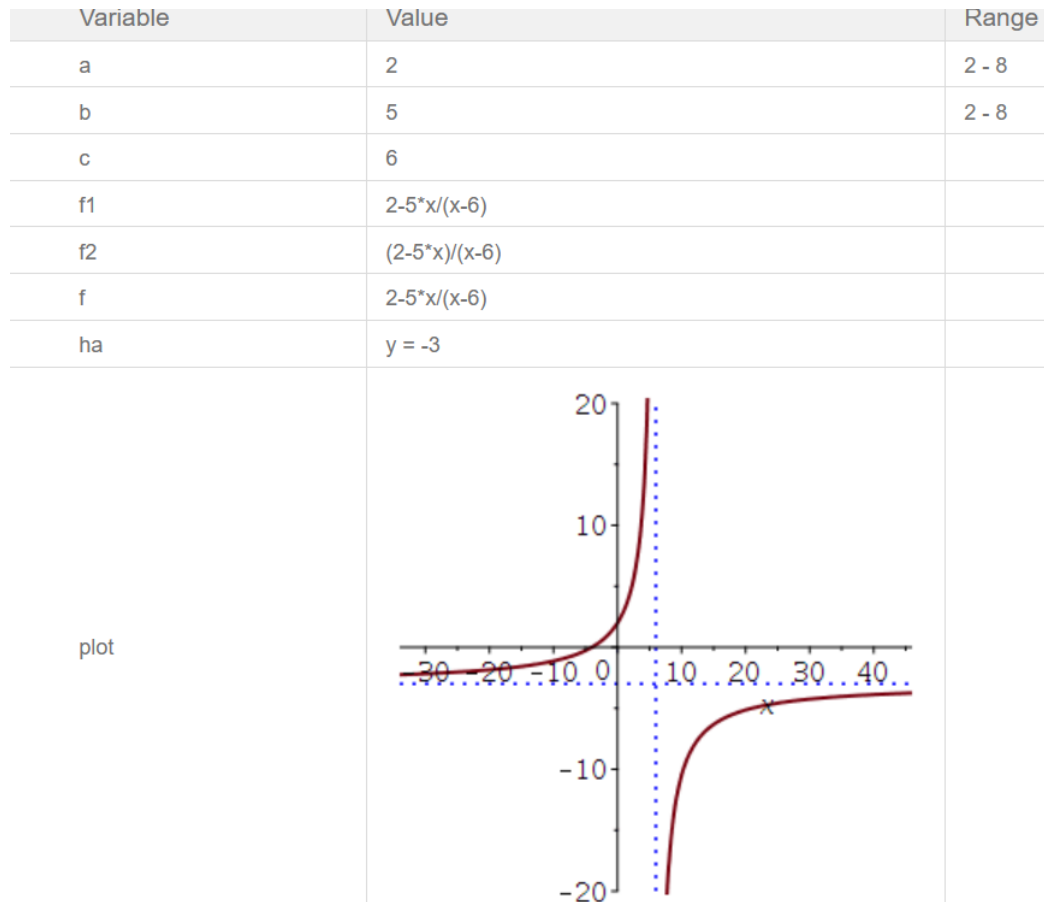


Figure 1.8: Verticale asymptoot

TIP: Met het plotcommando `plot([ $\$c$ ,  $t$ ,  $t=-20..20$ ])` maakt u een **parameterplot** waarvoor geldt $x = \$c$ en $y = t$ waarbij de parameter t loopt van -20 tot 20. Dat is in dit geval een handige manier om een verticale lijn te plotten. Een andere manier om een lijn door twee punten te tekenen is met het Maple-commando `line` uit het `plottools`-pakket. Zie ook in paragraaf (page 25). Zie ook in paragraaf (page 20) voor stuksgewijze functies met eindige discontinuïteiten.

1.1.1.8 Gekleurde gebieden in de grafiek

Het gebied onder de grafiek van een functie kunt u 'inkleuren' met de optie `filled=true` waarbij u ook de kleur en de transparantie kunt regelen. Zo zijn er ook verschillende gekleurde gebieden over elkaar heen te leggen.

1.1.1.8.1 Enkelvoudig gebied

In het volgende voorbeeld ziet u hoe het gebied onder een bergparabool boven de x -as gekleurd wordt waarbij de horizontale lijn door het zwaartepunt van dit oppervlak roodgestreept getekend is.

In het *Algorithm* programmeert u het volgende:

```
 $\$a = \text{switch}(\text{rint}(2), 2, 3);$ 
```

```

$y="$a^2-x^2";
$pardisplay=maple("printf(MathML[ExportPresentation](y=$y))");
$Sx=maple("int(1/2*($y)^2,x=-$a..$a)");
$opp=maple("int($y,x=-$a..$a)");
$yz=maple("$Sx/($opp)");
$plot=plotmaple("plots[display]([
plot($y,x=-$a..$a,scaling=constrained,filled=true,
color=tan,transparency=0.8),
plot($yz,x=-$a..$a,linestyle=dash,color=red)]),
plotoptions='height=300, width=300'");

```

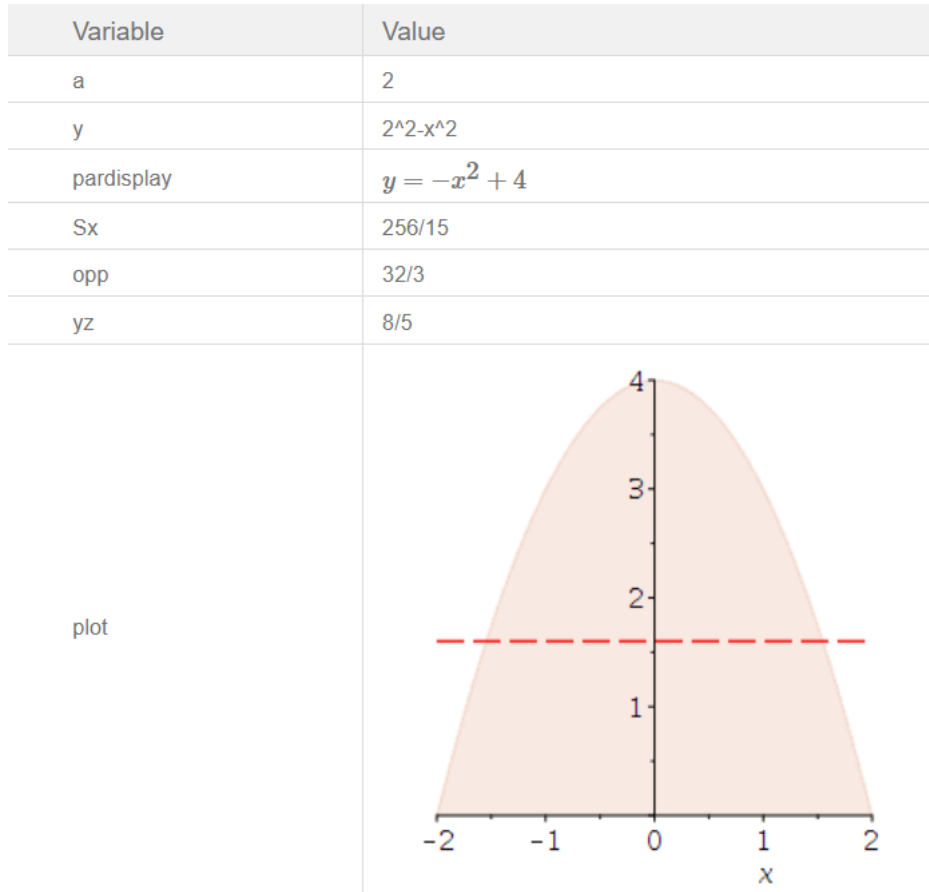


Figure 1.9: Eenvoudig gebied tussen functies kleuren

1.1.1.8.2 Meervoudig gebied

In het volgende voorbeeld is het gesloten gebied tussen een parabool, een rechte lijn en de x -as gekleurd.

Daarbij zijn ook de twee lijnen evenwijdig aan de assen door het zwaartepunt van het gesloten gebied in rood gestreept getekend. In het plotcommando is de lijst met de verschillende grafieken met het Maple-commando `display` uit het plotpakket over elkaar heen gelegd.

In het *Algorithme* programmeert u het volgende:

```

$y1="2-x";
$y2="x^2";
$x1="2-y";
$x2="sqrt(y)";

```

```

$Sx:=maple("int(($x1-($x2))*y,y=0..1)");
$opp:=maple("int($y2,x=0..1)+int($y1,x=1..2)");
$oppintdisplay="\displaystyle{\int_0^1($y1-$y2)\,\mathrm{d}y}=\displaystyle{\int_0^1($y2)\,\mathrm{d}x}+\displaystyle{\int_1^2($y1)\,\mathrm{d}x}=$opp\";
$yz:=maple("$Sx/($opp)");
$Sy:=maple("int($y2*x,x=0..1)+int(($y1)*x,x=1..2)");
$Sydisplay:=maple("printf(MathML[ExportPresentation](Int($y2*x,x=0..1)+Int(($y1)*x,x=1..2)=$Sy))");
$xz:=maple("$Sy/($opp)");
$plot:=plotmaple("plots[display]([
  plot($y2,x=0..1,scaling=constrained,filled=true,color=tan),
  plot($y1,x=1..2,scaling=constrained,filled=true,color=tan),
  plot($yz,x=0..2,linestyle=dash,color=red),
  plot([$xz,t,t=0..1.5],linestyle=dash,color=red)]),
  plotoptions='height=300, width=300');

```

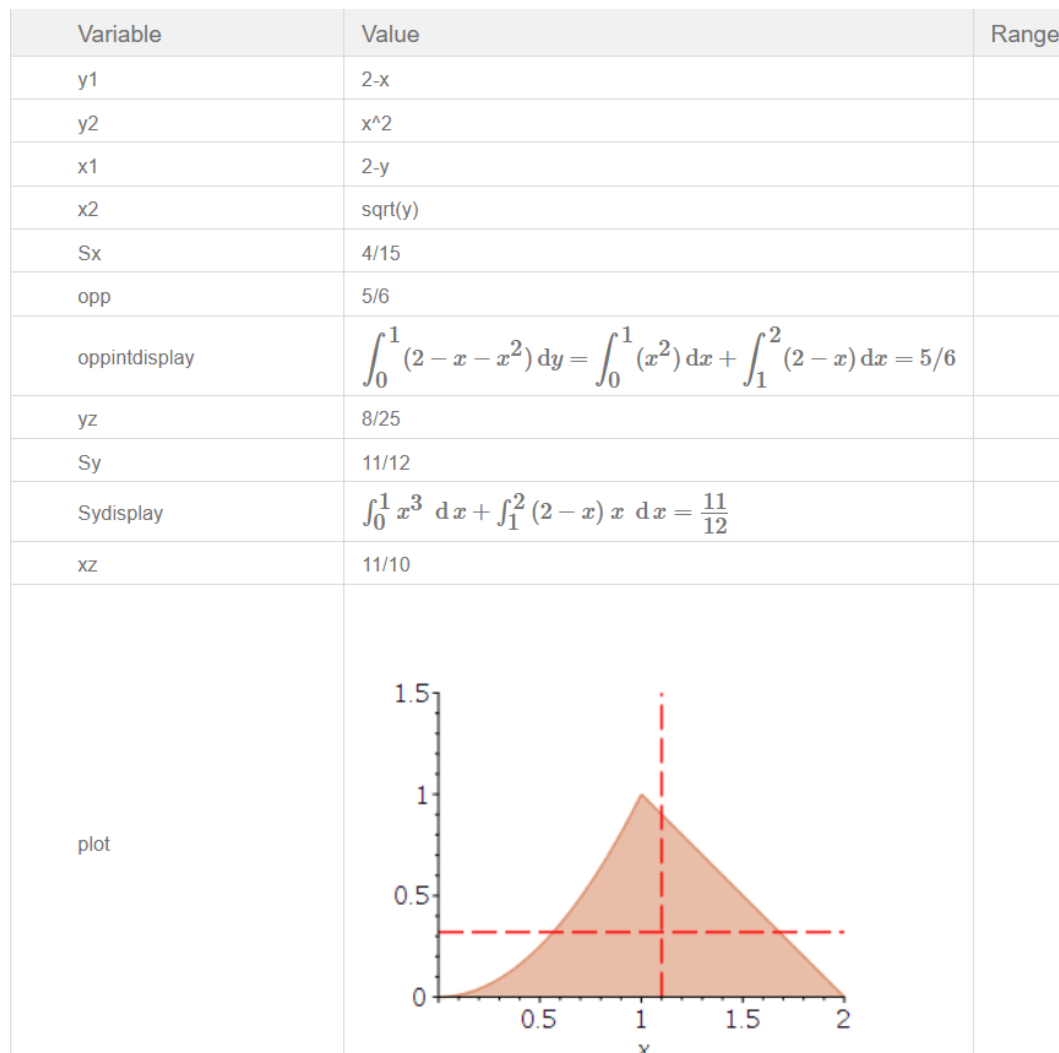


Figure 1.10: Meervoudig gebied tussen functies kleuren

1.1.1.8.3 Werken met transparantie

In het volgende voorbeeld ziet u dat de netto-oppervlakte onder de grafiek van de parabool gelijk is aan de oppervlakte van de donkerblauwe rechthoek zodat de gemiddelde waarde op het domein van de functie te berekenen is.

De gemiddelde waarde van de functie

$$f(x) = x^2 + 3x - 5$$

kan berekend worden met behulp van integraalrekening.

Beschouw de functie op het interval $[0, 4]$ en bekijk de grafiek.

Stel eerst de integraal op en bereken vervolgens deze integraal.

Dat doe je in twee stappen.

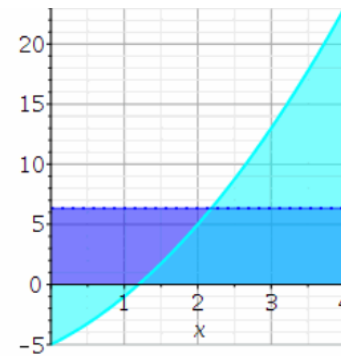


Figure 1.11: Gebieden kleuren met transparantie

Om de benodigde variabelen te krijgen programmeert u in het *Algoritme* het volgende:

```
$a=range(1, 3);
$b=range(-5, -1);
$c=range(4, 8);
$f=maple("x^2+($a)*x+($b)");
$interval=maple("[0, $c]");
$fdisplay=maple("printf(MathML[ExportPresentation](f(x)=$f))");
$fgem=maple("int($f, x=0..$c)/($c)");
$antw=decimal(3, $fgem);
$Fgem=maple(" printf(MathML[ExportPresentation](Int($f, x=0..$c)/
$c)) ");
$plot=plotmaple("
p1:=plot([$f, $fgem], x=0..$c, thickness=[2, 1],
linestyle=[1, 2], color=[cyan, blue]);
p2:=plot([$f, $fgem], x=0..$c, filled=true,
color=[$kleur, blue], transparency=0.5);
plots[display]({p1, p2}, gridlines=true),
plotoptions='height=250, width=250' ");
```

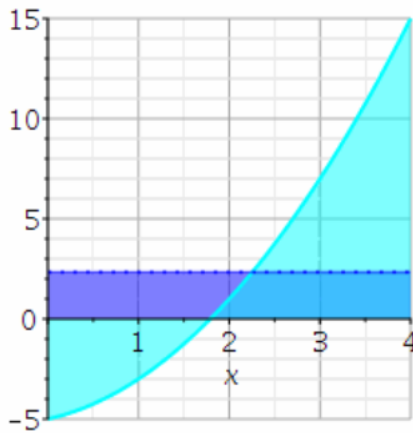
Variable	Value	Range
a	1	1 - 3
b	-5	-5 - -1
c	4	4 - 8
f	x^2+x-5	
kleur	cyan	
interval	[0, 4]	
fdisplay	$f(x) = x^2 + x - 5$	
fgem	7/3	
antw	2.333	
Fgem	$\frac{1}{4} \int_0^4 (x^2 + x - 5) dx$	
plot		

Figure 1.12: Algoritme van gebieden kleuren met transparantie

TIPS: Let eens op het commando om de variabele \$Fgem te definiëren waar er spaties tussen de quotes en het commando ingevoerd zijn. Het heeft als resultaat dat de onder- en bovengrens van de integraal mooier geplaatst worden in de formule.

Let ook op de dubbele punt als afsluiting van het declareren van de plots p1 en p2 binnen het maple-plotcommando.

De eerste plot p1 is het tekenen van de grafieken van de lijst met twee functies met verschillende kleur en lijnstijl.

De tweede plot p2 is de vulling van de gekleurde vlakken beide met transparantie 0.5 zodat ze beide te zien zijn.

Met behulp van display uit het plot-pakket legt u beide plots 'over elkaar heen'.

1.1.1.8.4 Gebieden afdekken met wit

Met de optie transparency=0 kunt u gekleurde gebieden gedeeltelijk afdekken. Het is dan wel belangrijk dat de verschillende grafieken die u over elkaar heen legt in de **goede volgorde** in een **lijst** genoemd staan (dus niet in een verzameling waar de volgorde niet van belang is).

Het 'over elkaar heen leggen' van de kleuren moet namelijk beslist in een bepaalde volgorde. De eerste grafiek in de lijst wordt het laatst er overheen gelegd. Als deze transparantie 0 heeft, dekt deze dus de andere grafieken af.

Programmeer het volgende in het *Algoritme* om de figuur te krijgen waarbij het gebied tussen sinus en cosinus gekleurd wordt.

```
$kleur=switch(rint(6), "navy", "khaki", "aquamarine", "maroon", "gold", "grey");
$f=maple("cos(x)");
$g=maple("sin(x)");
$plot=plotmaple("
p1:=plot($f, x=0..Pi/2, filled=true, color=white, transparency=0):
p2:=plot($g, x=0..Pi, color=$kleur, filled=true, transparency=0.3):
p3:=plot({$f, $g}, x=0..3/2*Pi, color=gray, thickness=2):
plots[display]([p1, p2, p3]),
plotoptions='height=250, width=250' ");
```

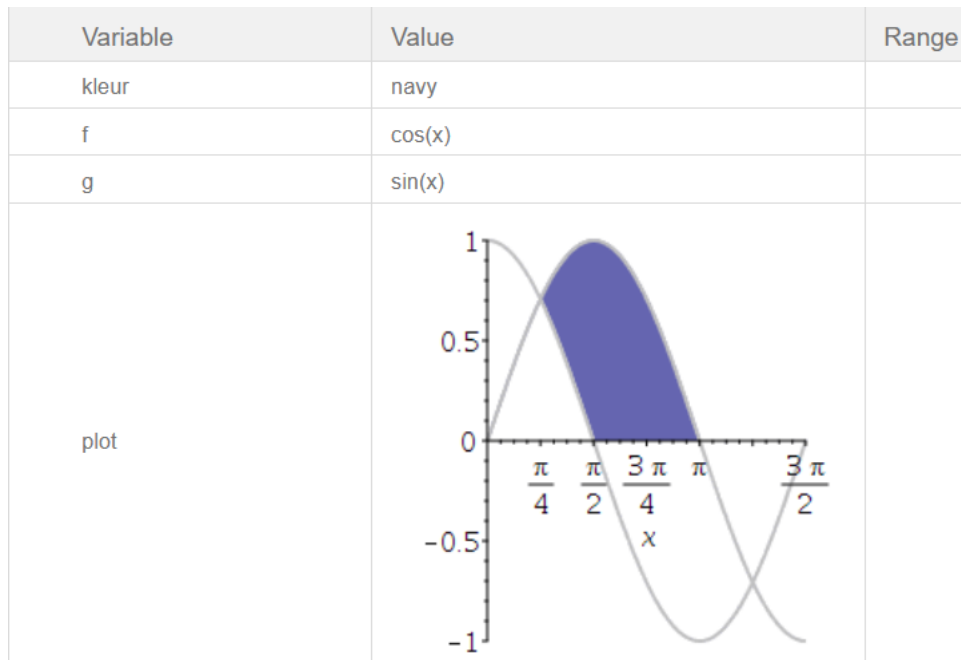


Figure 1.13: Oppervlakte afdekken met wit

TIP: Neem hier niet de optie `gridlines=true`, want anders worden de gridlines deels door het witte niet-transparante vlak afgedekt.

1.1.1.8.5 Gebieden tussen krommen van impliciete functies

Een impliciete kromme $F = 0$ verdeelt het vlak in twee delen ($F > 0$ en $F < 0$). Deze twee delen kunnen opgevuld worden met twee kleuren met de optie `coloring`.

In het volgende voorbeeld ziet u de grafieken van twee impliciete functies. Het gemeenschappelijk gebied dat daarmee omsloten wordt, kunt u zo een bepaalde kleur geven door gebruik te maken van transparantie.

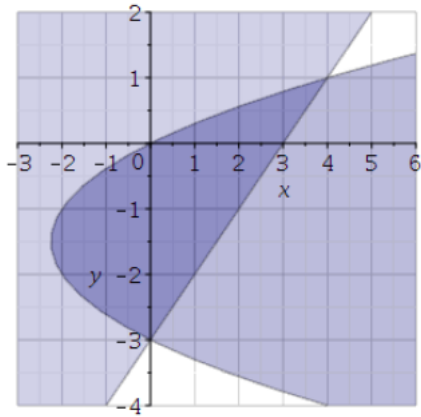


Figure 1.14: Kleuring van gebieden met impliciete functies

De figuur maakt u in het *Algorithm* met de volgende programmeerregels:

```
$a=range(1,3);
$b=range(1,3);
$kleur=switch(rint(6),"navy","khaki","aquamarine","maroon","gold","grey");
$f="y^2+($a)*y";
$g="($b)*y+($b)*($a)";
$plot=plotmaple("
p1:=plots[implicitplot]([x-($f)=0],x=-3..($b^2+$a*$b+2),
y=-($a)-1..($b)+1,coloring=[white,$kleur],
transparency=0.7,filledregions=true):
p2:=plots[implicitplot]([x-($g)=0],x=-3..($b^2+$a*$b+2),
y=-($a)-1..($b)+1,coloring=[$kleur,white],
transparency=0.7,filledregions=true):
plots[display]([p1,p2],gridlines=true),
plotoptions='height=300,width=300'");
```

1.1.1.8.6 Gebieden kleuren bij (lineaire) ongelijkheden

Een alternatief voor de figuur hierboven met de overlappende gebieden van grafieken tussen impliciete krommen is gebruik te maken van het Maple-commando `inequal` uit het `plots`-pakket.

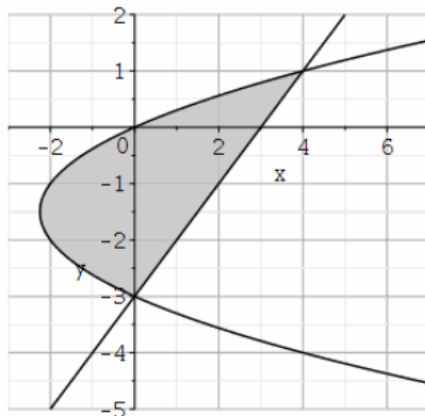


Figure 1.15: Kleuren van gebieden bij lineaire ongelijkheden

Het is dan mogelijk om met een verzameling ongelijkheden (of een lijst met ongelijkheden) een grafiek te maken en de ingesloten gebieden die aan de ongelijkheden voldoen (feasible regions) te kleuren.

Bij het *Algorithm* van het vorige voorbeeld kunt u de alternatieve figuur programmeren met de volgende regels:

```
$plot1=plotmaple("
plots:-inequal([x>=y^2+($a)*y, x<=($b)*y+($b)*($a)],
x=-3..($b^2+$a*$b+3), y=-$a-2..$b+1, gridlines=true,
optionsfeasible=(color=gray, transparency=0.2)),
plotoptions='height=300, width=300'");
```

TIP: Deze manier met het Maple commando `inequal` is goed te doen als de ongelijkheden lineair zijn. Bij andere dan lineaire ongelijkheden, zoals hier duurt het even voordat de grafiek gegenereerd is.

1.1.1.9 Grafieken van stuksgewijze functies

Functies die gedefinieerd zijn met verschillend voorschrift op verschillende intervallen zijn te visualiseren door ze goed te definiëren en dan gewoon te plotten.

Er zijn drie situaties om zo'n functie te beschrijven:

- Met het Maple commando `PIECEWISE`
- Met behulp van de stapfunctie met Maple-commando `Heaviside`
- Een absolute-waarde functie omzetten in een stuksgewijze functie

1.1.1.9.1 De stuksgewijze functie

Met de volgende programmaregels in het *Algorithm* hebt u een paar voorbeelden van stuksgewijze functies die met `PIECEWISE` zijn gedefinieerd.

```
$f=maple("PIECEWISE([4,t<-2],[t^2,t>=-2 and t<=3],[1,3<t])");
$fdisplay=maple("printf(MathML:-ExportPresentation(y(t)=$f))");
$fplot=plotmaple("plots[display](
{plot($f,t=-5..5, linestyle=2), plot($f,t=-5..5, discontinuity=true)}),
plotoptions='height=150, width=150'");
$g=maple("PIECEWISE([0,t<-2],[(-t-1)^2,t>=-2 and t<=3],[1,3<t])");
$gdisplay=maple("printf(MathML:-ExportPresentation(y(t)=$g))");
$gplot=plotmaple("plots[display](
{plot($g,t=-5..5, linestyle=2), plot($g,t=-5..5, discontinuity=true)}),
plotoptions='height=150, width=150'");
$h=maple("PIECEWISE([0,t<-2],[(-t-1)^2-3,t>=-2 and t<=3],
[1,3<t])");
$hdisplay=maple("printf(MathML:-ExportPresentation(y(t)=$h))");
$hplot=plotmaple("plots[display](
{plot($h,t=-5..5, linestyle=2), plot($h,t=-5..5, discontinuity=true)}),
plotoptions='height=150, width=150'");
$k=maple("PIECEWISE([4,t<0],[-(-t-1)^2+5,t>=-2 and t<=3],
[1,3<t])");
$kdisplay=maple("printf(MathML:-ExportPresentation(y(t)=$k))");
$kplot=plotmaple("plots[display](
{plot($k,t=-5..5, linestyle=2), plot($k,t=-5..5, discontinuity=true)}),
plotoptions='height=150, width=150'");
```

Met een tabel hebt u het functievoorschrift en de grafiek naast elkaar.

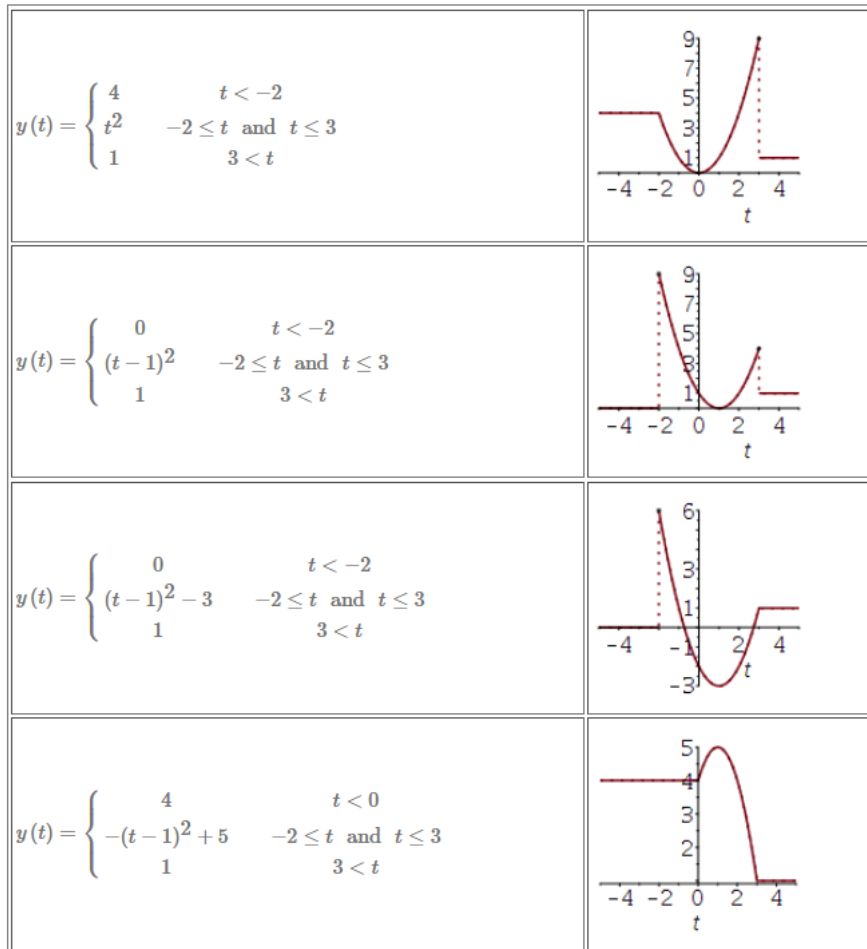


Figure 1.16: Stuksgewijze functie

TIPS:

De betekenis van `PIECEWISE([4, t<-2], [t^2, t>=-2 and t<=3], [1, 3<t])` is dat tussen de vierkante haken de drie verschillende functies op de verschillende trajecten gedefinieerd worden. Het middelste interval is $[-2, 3]$.

Voor dezelfde figuur had hier ook met de volgende opdracht kunnen volstaan

`PIECEWISE([4, t<-2], [t^2, t<=3], [1, t>3])`

De opdracht wordt namelijk van links naar rechts gelezen.

Zie in het *Algorithm* hoe de twee grafieken van de functie met en zonder de optie `discont=true` over elkaar heen gelegd worden met verschillende lijnstijlen.

Voor meer informatie over verschillende functies in één grafiek, zie paragraaf (page 4).

1.1.1.9.2 Met behulp van de stapfunctie (Heaviside)

Met de volgende regels programmeert u in het *Algorithm* een voorbeeld van een stuksgewijze grafiek.

```
$b=switch(rint(7), -3, -2, -1, 1, 2, 3, 4);
$c=switch(rint(7), -3, -2, -1, 1, 2, 3, 4);
condition:gt($c,$b);
$f=maple("t*Heaviside(t-($b))-t*Heaviside(t-($c))");
$fdisplay=maple("printf(MathML:-ExportPresentation( $f))");
$plot=plotmaple(" plots[display]([
```

```
plot($f,t=$b-4..$c+1,color=navy,thickness=3,scaling=constrained,
labels=[t,f],gridlines=true,discont=true),
plot($f,t=$b-4..$c+1,color=navy,linestyle=2)],
plotoptions='height=200, width=350'");
```

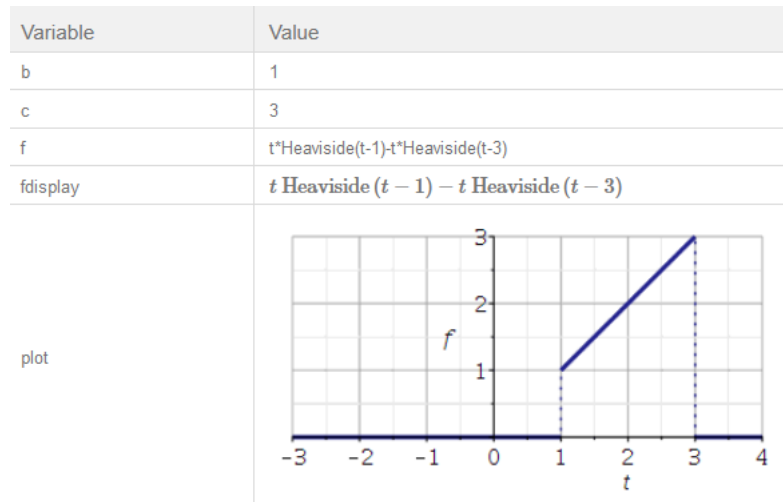


Figure 1.17: Stuksgewijze functie met de stapfunctie

TIP: Hier zijn weer twee grafieken over elkaar heen gelegd: een met `discont=true` en een gestippelde zonder optie `discont=true`.

1.1.1.9.3 Absolute waarde-functie

In het volgende voorbeeld wordt een absolute waarde-functie geplot. Dat kan gewoon met het `plot`commando in Maple.

Preview

Gegeven is de absolute waardefunctie $y = |x^2 - 9| + |2x - 10|$

Met de grafiek.

Deze functie wil je schrijven als een functie in stukken.

Het aantal intervallen waarop je de functie moet verdelen is

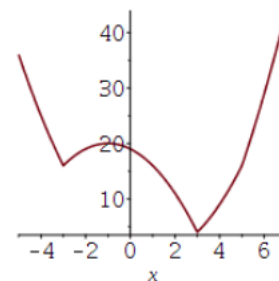


Figure 1.18: Absolute waarde-functie

In het volgende *Algorithme* staan de programmeringsregels voor het plotten van de figuur van de absolute waarde-functie.

```
$gx12=range(1,3);
$gx3=switch(rint(3),4,5,6);
$ga="abs(x^2-($gx12)^2)+abs(2*x-2*$gx3)";
$gadisplay=maple("printf(MathML:-ExportPresentation(y=$ga))");
$gplot=plotmaple("plot($ga,x=-$gx12-2..$gx3+2),
plotoptions='height=200, width=200'");
```

TIPS:

Een dergelijke functie is om te zetten in een stuksgewijze functie, maar eventueel ook in een functie beschreven met behulp van de Heaviside functie.

In het *Algorithm* programmeert u dat met de volgende Maple-opdrachten waarbij geheel automatisch de stuksgewijze definitie van de functie verschijnt en ook de definitie met behulp van Heaviside.

```
$gp:=maple("convert($ga, piecewise)");
$gpdisplay:=maple("printf(MathML:-ExportPresentation(y=$gp))");
$gp1:=maple("convert($ga, Heaviside)");
$gp1display:=maple("printf(MathML:-ExportPresentation(y=$gp1))");
```

gp	PIECEWISE([x^2-2*x+1, x <= -3],[-x^2-2*x+19, x < 3],[x^2-2*x+1, x < 5],[x^2+2*x-19, 5 <= x])
gpdisplay	$y = \begin{cases} x^2 - 2x + 1 & x \leq -3 \\ -x^2 - 2x + 19 & x < 3 \\ x^2 - 2x + 1 & x < 5 \\ x^2 + 2x - 19 & 5 \leq x \end{cases}$
gp1	$-2x^2 \text{Heaviside}(x+3) + x^2 + 2x^2 \text{Heaviside}(x-3) + 4 \text{Heaviside}(x-5)x + 1 + 18 \text{Heaviside}(x+3) - 18 \text{Heaviside}(x-3) - 20 \text{Heaviside}(x-5)$
gp1display	$y = -2x^2 \text{Heaviside}(x+3) + x^2 + 2x^2 \text{Heaviside}(x-3) + 4 \text{Heaviside}(x-5)x + 1 + 18 \text{Heaviside}(x+3) - 18 \text{Heaviside}(x-3) - 20 \text{Heaviside}(x-5)$

Figure 1.19: Omzetten naar een stuksgewijze functie

1.1.1.10 Logaritmische schalen

Er zijn twee belangrijke manieren om logaritmische schaalverdeling te maken voor een grafiek.

- Enkellogaritmische schaal
- Dubbellogaritmische schaal

1.1.1.10.1 Enkellogaritmische schaal

Een voorbeeld met logaritmische schaal langs alleen de verticale as terwijl de horizontale as lineair is, ziet u in het volgende voorbeeld van een exponentiële functie.

Preview

In de figuur zie je de grafiek van een exponentiële functie $y = b \cdot g^t$ getekend met op de verticale as de logaritmische schaal.

Lees de waarde van b af in de grafiek:

$b =$

Lees af in de grafiek en bereken de waarde van g :

$g =$ (rond af op 3 decimalen)

Bepaal ook de groeiconstante k :

$k =$ (rond af op 3 decimalen)

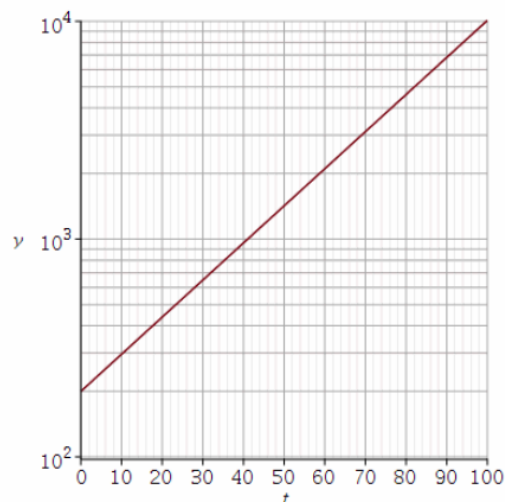


Figure 1.20: Enkellogaritmische schaal

Om de grafiek te maken programmeert u de volgende regels in het *Algorithm*:

```
$g=switch(rint(5),1.045,1.028,1.036,1.040,1.055);
$b=switch(rint(5),100,200,300,400,500);
```

```
$p=plotmaple("plot($b*$g^t,t=0..100,y=100..10000,
axis[2]=[mode=log,gridlines=20],
axis[1]=[gridlines=[51,majorlines=5]],tickmarks=[10,3]),
plotoptions='height=400, width=400'");
$K=maple("evalf[4](ln($g))");
$k=numfmt("0.000",$K);
```

Met extra opties voor de assen en aantal tickmarks en gridlines, kunt u de figuur aanpassen.

TIP: Zorg ervoor dat de figuur niet te klein wordt afgebeeld als er dingen uit afgelezen moeten worden door de student.

Let ook eens op de aanpassing van de variabele \$k waarbij het numerieke format wordt afgedwongen. (Het antwoord komt anders met de wetenschappelijke notatie in de feedback.)

1.1.1.10.2 Dubbellogaritmische schaal

Een voorbeeld met logaritmische schaal langs de verticale én de horizontale as ziet u in het volgende voorbeeld van een machtsfunctie.

Preview

De grafiek van de functie $f(x) = bx^a$ is een rechte lijn als de beide assen logaritmische schaal zijn. Bereken de parameters a en b van deze functie door de grafiek af te lezen.

$a =$ (rond af op 2 decimalen)

$b =$ (rond af op 2 decimalen)

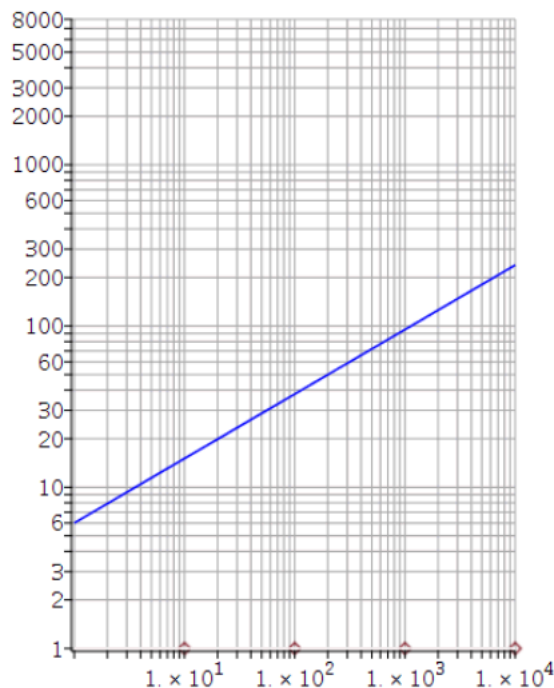


Figure 1.21: Dubbellogaritmische schaal

Om de grafiek te maken programmeert u de volgende regels in het *Algorithm*:

```
$b=range(1,6);
$a=decimal(2,range(0.35,0.85,0.05));
$f=maple("$b*x^$a");
$p=plotmaple("plots[display]({
plot([1,$f],x=1..10000,y=1..8000,color=[white,blue],
axis[1]=[mode=log],
axis[2]=[mode=log,gridlines=true,tickmarks=[30,30]],
plot([[10,1],[100,1],[1000,1],
[10000,1]],style=point,symbolsize=15)
}),plotoptions='height=500, width=400'");
```

TIP: In deze figuur is nog extra een grafiek met punten op de horizontale as meegegeven. Met het Maple-commando `display` uit het `plot`-pakket legt u de verschillende grafieken 'over elkaar' heen.

1.1.1.11 Plottools

Met het `plottools`-pakket in Maple zijn er veel figuren gemakkelijk te plotten als alternatief voor het plotten van functies.

U moet daarbij denken aan veelhoeken (ook driehoeken), pijlen, cirkels (disk voor een opgevulde cirkel), bogen en sectoren.

Het `Plottools`-pakket biedt ook commando's om gemakkelijk ruimtefiguren te plotten zoals cilinders en bollen en veelvlakken.

U ziet hier een paar voorbeelden van figuren in het platte vlak, eventueel aangevuld met andere figuren en opties.

Een figuur wordt met een commando uit het `plottools`-pakket gedefinieerd en met behulp van `plots[display]` in een grafiek gezet. Zo kunt u meer verschillende onderdelen definiëren en in één figuur samenbrengen.

1.1.1.11.1 Polygoon

In de volgende figuur ziet u een veelhoek. Met behulp van het Maple-commando `polygoon` uit het `plottools`-pakket is een willekeurige veelhoek getekend. U kunt zoveel hoeken nemen als u zelf wilt. In een lijst staan alle hoekpunten genoemd.

Preview

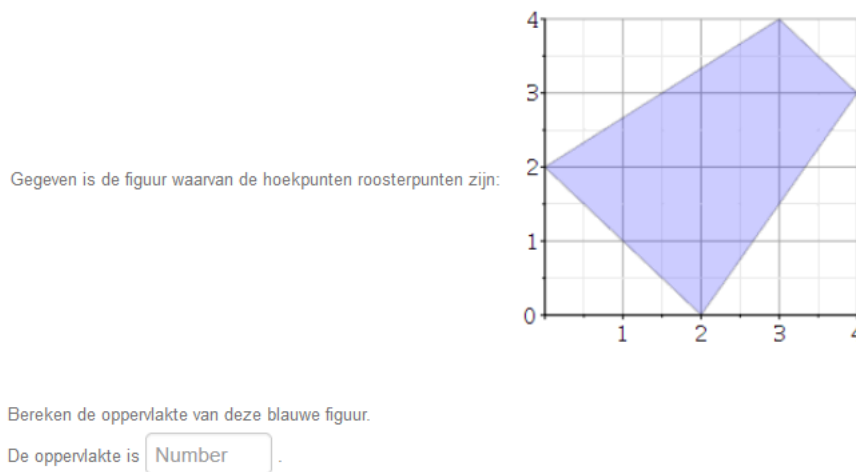


Figure 1.22: Figuur met polygoon en gridlines

In het *Algorithm* programmeert u het volgende:

```
$b1=range(0,4);
$antw=7-1/2+($b1)/2;
$q=plotmaple("plots[display](plottools[polygoon]([[0,$b1], [3,4],
[4,3], [2,0]],color=blue,transparency=0.8),gridlines=true),
plotoptions='height=250, width=250'");
```

Hierboven is met de enkele optie `gridlines=true` niet veel aan te passen aan de standaard-weergave van de gridlines.

Met de optie `transparency` is de doorzichtigheid van het gekleurde oppervlak wat in te stellen met een waarde tussen 0 en 1.

1.1.1.11.2 Driehoeken en tekst in een cirkel

In het volgende voorbeeld ziet u twee driehoeken ABC en MBD ingeschreven in een cirkel met middelpunt M.

Punt D is zó op de cirkel geplaatst dat de zijden AC en MD elkaar loodrecht snijden.

De letters erbij gaan met het Maple-commando `Textplot` uit het plotpakket.

Het *Algorithm* om de figuur te maken is als volgt:

```
$A:=range(20, 70, 2);
$M1=90-$A;
$M2=180-$M1;
$B=(180-$M2)/2;
$p:=plotmaple("with(plottools):with(plots):
c:=circle([0,0],1):
p1:=polygon([[ -1,0],[1,0],[cos(2*$A*Pi/180),sin(2*
$A*Pi/180)]] ,color=blue,transparency=0.9):
p2:=polygon([[0,0],[1,0],
[cos($M2*Pi/180),sin($M2*Pi/180)]] ,color=blue,transparency=0.9):
t:=textplot([[ -1,0,A],[0,-0.2,M],[1,0,B],[cos(2*$A*Pi/180),sin(2*
$A*Pi/180),C],[cos($M2*Pi/180),sin($M2*Pi/180),D]],
align=[right,above],font=[Times,Italic,16]):
display({c,p1,p2,t},scaling=constrained,axes=none),
plotoptions='height=300, width=300'");
```

Variable	Value	Range
A	54	20 - 70
M1	36	
M2	144	
B	18	

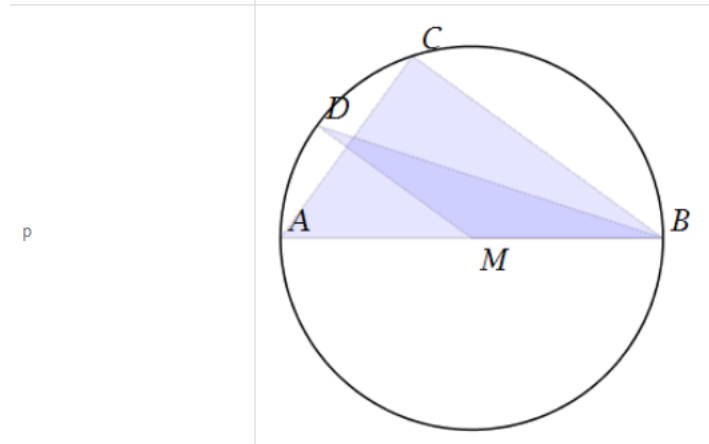


Figure 1.23: Inggeschreven driehoeken

TIP: Binnen de opdracht `plotmaple` worden hier eerst met `with(plottools):with(plots):` de pakketten `plottools` en `plots` geactiveerd zodat de commando's `polygon` en `circle` uit het `plottools`-pakket, en `textplot` en `display` uit het `plot`-pakket direct zijn te gebruiken.

1.1.1.11.3 Sector

In dit voorbeeld ziet u een combinatie van het gekleurde gebied gemaakt met het Maple-commando `inequal` uit het `plot`-pakket (zie paragraaf (page 19)) en verder nog een gekleurd gebied met het Maple-commando `sector` uit het `plottools`-pakket.

De verzameling complexe getallen die voldoet aan de voorwaarden dat de modulus tussen 1 en 2 ligt en waarvan het argument tussen $\frac{1}{4}\pi$ en $\frac{3}{4}\pi$ ligt is te visualiseren met het volgende

Algorithm.

De verzameling is met LaTeX in een formule geschreven en de variabele \$p1 is de bijbehorende figuur.

```
$BV1="y>x";
$BV2="y>-x";
$verz1="\(\{z \in \Bbb{C}\} \mid 1 \leq |z| \leq 2 \wedge \frac{1}{4}\pi \leq \text{Arg}(z) \leq \frac{3}{4}\pi\)\)";
$p1=plotmaple("plots[display]([plots[inequal]({$BV1,
$BV2}, x=-2..2, y=-2..2, gridlines=true, optionsfeasible=[color=yellow, transparency=0.8],
plottools[sector]([0,0],1,Pi/4..3*Pi/4,color=white),
plottools[sector]
([0,0],2,Pi/4..3*Pi/4,color=blue,transparency=0.6)],
labels=[`Re-as`, `Im-as`], labeldirections=[horizontal,vertical]),
plotoptions='height=200, width=200'");
```

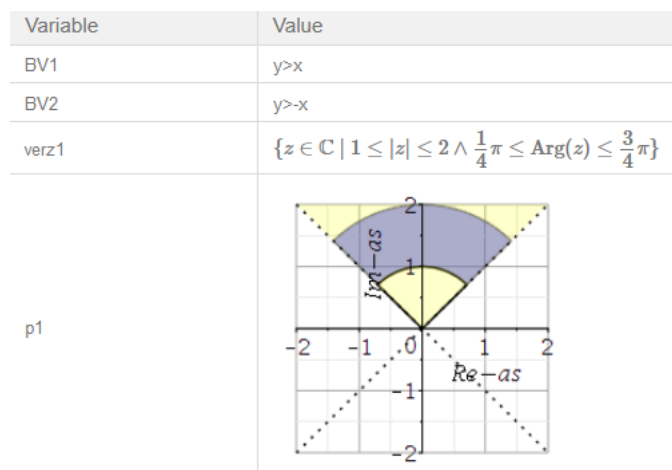


Figure 1.24: Gekleurd gebied met sectoren

TIP: U ziet hier weer dat een aantal plots 'over elkaar heengelegd zijn' met behulp van het Maple-commando `display` uit het plot-pakket.

Let wel op dat u de verschillende plots **in een lijst** opnoemt, want de volgorde is van belang bij het afdekken van de witte sector (met transparantie 0) over een deel van de grijze sector. Zie ook paragraaf (page 17).

1.1.1.11.4 Cirkel, vierkant en lijnstuk

In dit voorbeeld ziet u een aantal figuren die met Maple gemaakt zijn met behulp van commando's uit het plottools-pakket: een cirkel (met `disk`), een vierkant (met `polygon`) en een lijn (met `line`) in het volgende *Algorithm*:

```
$d=switch(rint(5),4,6,8,10,12);
$p=plotmaple("cirk1:=plottools[disk]([0,0],1,color=white):
l3:=plottools[line]([-1,0],[1,0],color=blue):
vk:=plottools[polygon]([[-1,-1],[1,-1],[1,1],
[-1,1]],color=blue,transparency=0.7):
tvk:=plots[textplot]([0,0.1,type(d=$d)],font=[Times,12]):
plots[display]([cirk1,vk,l3,tvk],scaling=constrained,axes=none),
plotoptions='height=200, width=200'");
```

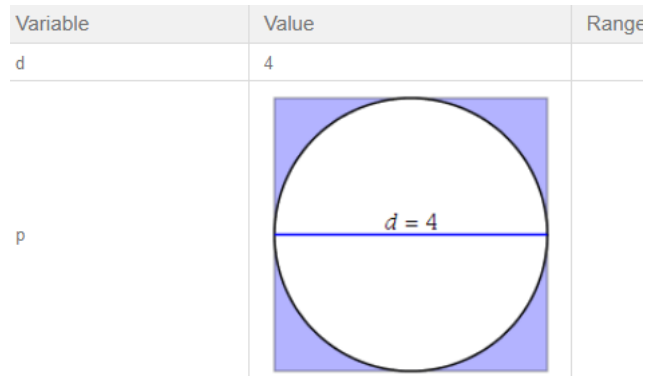


Figure 1.25: Cirkel, vierkant en lijnstuk

TIP: Binnen de Maple-opdracht zijn drie figuren gedefinieerd. Let weer op de volgorde van de lijst van deze figuren die over elkaar heengelegd zijn met het Maple-commando `display` uit het plotpakket met het oog op het afdekken van de figuren onderling.

TIP: Let ook op de tekst in de figuur gemaakt met het Maple-commando `textplot` uit het plotpakket.

1.1.1.12 Plottools 3D

Voor sommige figuren in 3D zoals cilinders, bollen, pyramides en dergelijke is het `plottools`-pakket erg geschikt.

U definieert in dit pakket de onderdelen die u vervolgens samenvoegt in één figuur met behulp van `display` uit het `plots`-pakket.

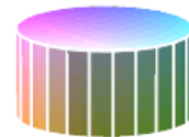
1.1.1.12.1 Cilinder met `plottools`

In dit voorbeeld maakt u een plaatje van een cilinder (`cylinder`) op schaal. Met de gegevens moet de student de inhoud berekenen.

Van een cilinder is gegeven dat de totale oppervlakte inclusief bodem en deksel gelijk is aan 90.

Verder is nog gegeven dat de hoogte 0.5 maal de diameter is van het grondvlak.

Bereken de inhoud van de cilinder.



De inhoud van de cilinder is (rond af op 1 decimaal)

Figure 1.26: Cilinder met `plottools`

Hieronder ziet u het *Algorithm* om de figuur van deze variabele cilinder te maken.

```
$f=switch(rint(2), decimal(1, rand(0.5, 0.9)), decimal(1, rand(1.2, 2)));
$p=plotmaple("plots[display](plottools[cylinder]([0, 0, 0], 1, 2*$f),
scaling=constrained, axes=none), plotoptions='height=200,
width=200'");
$0=range(40, 120, 10);
$r=maple("max(fsolve(2*Pi*r^2+2*Pi*r*$f*2*r=$0, r))");
```

```

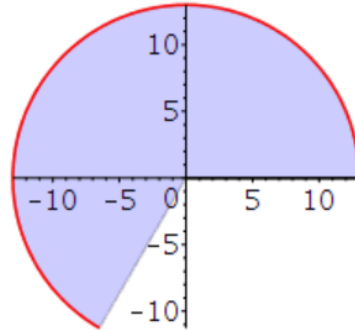
$rr=decimal(1,$r);
$h=2*$r*$f;
$V=maple("Pi*$r^2*$h");
$inhoud=decimal(1,$V);

```

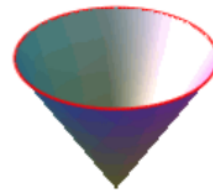
1.1.1.12.2 Kegels met plottools

In het volgende voorbeeld is een kegel (cone) gemaakt met plottools

Een cirkelsector, gesneden uit een cirkel met straal 13, heeft als middelpuntshoek een hoek van $\frac{4}{3}\pi$ radialen



Deze sector vouw je tot een kegel.



Bereken de totale oppervlakte van de kegel inclusief deksel.

De totale oppervlakte van de kegel is (rond af op 1 decimaal)

Figure 1.27: kegel met plottools

Het bijbehorende Algorithm is:

```

$alfa=switch(rint(5),"3/2*Pi","Pi","4/3*Pi","2/3*Pi","5/4*Pi");
$alfadisplay=maple("printf(MathML[ExportPresentation]($alfa))");
$R=range(6,15);
$r=maple("$alfa*$R/(2*Pi)");
$rdisplay=maple("printf(MathML[ExportPresentation]($r))");
$h=maple("sqrt($R^2-$r^2)");
$hdisplay=maple("printf(MathML[ExportPresentation]($h))");

$p1=plotmaple("plots[display]({plottools[sector]([0,0],$R,0..
$alfa,
color=blue,transparency=0.8)
,plottools[arc]([0,0],$R,0..$alfa,color=red)
,plottools[line]([0,0],
[$R,0],color=black)},scaling=constrained),plotoptions='height=200,
width=200'");

```

```

$p2=plotmaple("plots[display]({plottools[cone]([0,0,0],$r,
$h,style=patchnogrid)
,plots[spacecurve]([$r*cos(t),$r*sin(t),
$h],t=0..2*Pi,color=red,filled=true)},scaling=constrained,axes=none,orientation=[-120,60],width=200'");

$p3=plotmaple("plots[display]({plottools[cone]([0,0,0],$r,
$h,style=patchnogrid)
,plots[implicitplot3d](x=0,x=-$r..$r,y=-$r..$r,z=0..$h)},
scaling=constrained,orientation=[-120,60]),plotoptions='height=200,
width=200'");

$p4=plotmaple("plots[display]({plottools[polygon]([0,0],[$r,$h],
[-$r,$h]),color=blue,transparency=0.8)
,plottools[line]([- $r,$h],[$r,$h],color=red)
,plottools[line]([0,0],[$r,$h],color=black)},scaling=constrained),plotoptions='height=150,
width=150'");

$Inhoud=maple("1/3*Pi*$r^2*$h");
$inhoud=decimal(1,$Inhoud);
$inhouddisplay=maple("printf(MathML[ExportPresentation]
($Inhoud))");
$Opp=maple("Pi*$r^2+1/2*$alpha*$R^2");
$sopp=decimal(1,$Opp);

```

TIP: Ten behoeve van de feedback zijn in het *Algorithm* nog twee extra grafieken geprepareerd namelijk \$p3 en \$p4.

In \$p3 ziet u hoe er een plottools object te samen met een vlak in 3D als impliciete functie in één figuur zijn samengebracht.

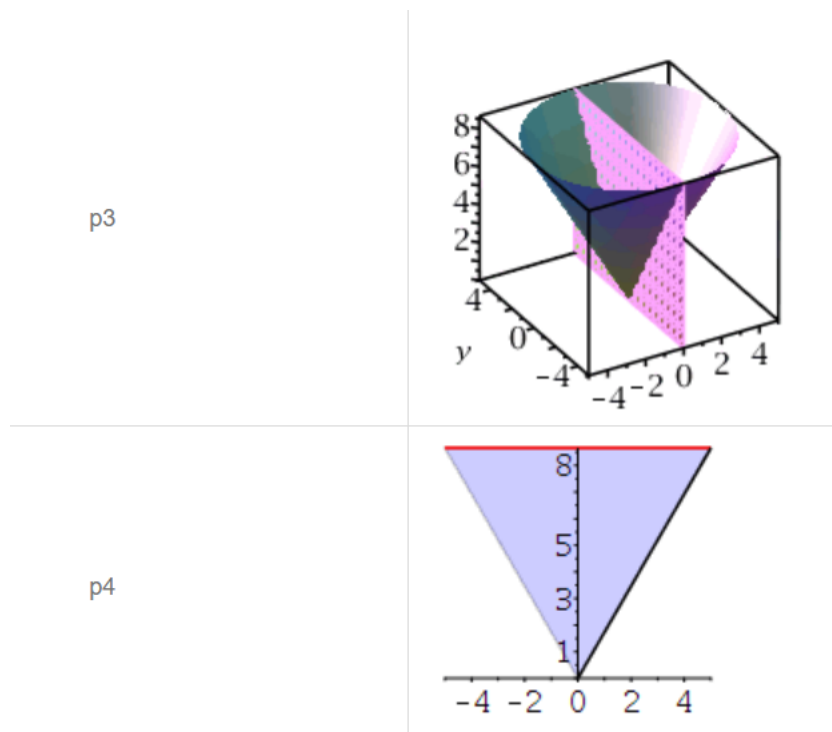


Figure 1.28: Een cilinder en een vlak samen

1.1.1.13 Plottools vectoren

Grafieken met vectoren zijn gemakkelijk te maken met Maple. Berekeningen en figuren kunnen in het *Algorithm* volledig voorbereid worden.

Een pijl kunt u maken door *beginpunt* [1,0] of *beginvector* <0,2> mee te geven, en een *eindpunt* in de vorm van een *punt* óf een *richting* in de vorm van een *Vector*.

Dan een rijtje met attributen(eigenschappen in de vorm van getallen), achtereenvolgens: de dikte van de pijlpoot, de dikte van de punt en de verhouding van lengte van de punt tot de lengte van de gehele pijl.

Der pijl wordt omrand door een zwarte contour (border). Meestal wilt u die contour niet en die maakt u dan met dezelfde kleur als de pijl.

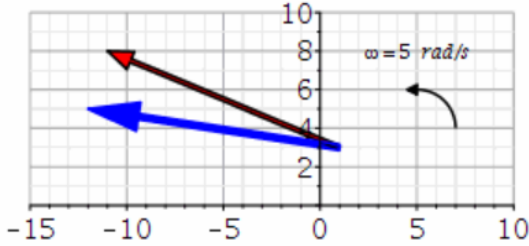
Variable	Value
w	5
commentaar	wb width of the body of the arrow wh width of the head of the arrow hh height of the head of the arrow as a ratio of the length of the body Punt en richtingsvector of begin en eindpunt.
fig2	

Figure 1.29: Vectoren met plottools

Het *Algorithm* is als volgt:

```
$w:=range(5,10,1);
$commentaar="wb = width of the body of the arrow <br/>
wh = width of the head of the arrow<br/>
hh = height of the head of the arrow as a ratio of the length of
the body<br/>
Beginpunt en richtingsvector of beginpunt en eindpunt.";
$fig2=plotmaple("
p1:=plottools[arrow]([1,3],
[-12,5],0.3,1.5,0.2,color=blue,border=[color=blue]):
p1a:=plottools[arrow]([1,3],<-12,5>,0.2,1,0.1,color=red):
p2:=plottools[arc]([5,4],2,0..Pi/2):
p3:=plottools[arrow]([5,6],[4.5,6],0,0.5,1):
p4:=plots[textplot]([5,8,typeset(omega=$w,`rad/
s`)],font=[Roman,8]):
plots[display]({p1,p1a,p2,p3,p4},scaling=constrained,
gridlines=true,view=[-15..10,0..10]),plotoptions='height=150,
width=300'");
```

1.1.1.14 Geometrie

Met behulp van het geometry-pakket in Maple zijn handig meetkundige berekeningen te doen waarbij u de figuur ook weer dynamisch presenteert met letters bij de meetkundige figuur.

Gegeven is een rechthoekige driehoek ABC .

De hoogtelijn CH is ook getekend in de figuur.

Daarin zijn verder nog gegeven dat $AB = 152$ en $BC = 67.98$.



Gevraagd wordt om de lengte van CH te berekenen.

$CH =$ (rond af op 2 decimalen)

Figure 1.30: title of the figure

In het *Algorithm* definieert u alles wat nodig is voor de figuur. De meetkunde met de maatvoering is dan ook meteen vastgelegd.

Met behulp van het commando `draw` in dit geometry-pakket komen de voorgedefinieerde onderdelen bij elkaar in één figuur.

```
$a:=range(1,4);
$x:=decimal(1,rand(0.5,2*$a-0.5));
$y1:=maple("max(fsolve(($x-$a)^2+y^2=$a^2,y))");
$y2:=maple("min(fsolve(($x-$a)^2+y^2=$a^2,y))");
$y:=switch(rint(2),$y1,$y2);
$driehoek:=plotmaple("with(geometry):
triangle(T,[point(A,0,0),point(B,2*$a,0),point(C,$x,$y)]):
altitude(hC,C,T,H):
draw([T(filled=true,color=gray),A,B,C,hC,H],scaling=constrained,axes=none,printtext=true),plotoptions='he
width=350' ");
$f:=range(2,20);
$AB=2*$a*$f;
$Geglijst:=maple("with(geometry):triangle(T,[point(A,0,0),point(B,2*$a,0),point(C,
$x,$y)]):altitude(hC,C,T,H):[distance(A,C),distance(B,C)]");
$Gevrlijst:=maple("with(geometry):triangle(T,[point(A,0,0),point(B,2*$a,0),point(C,
$x,$y)]):altitude(hC,C,T,H):[distance(A,H),distance(B,H),distance(C,H)]");
$index1:=range(1,2);
$index2:=range(1,3);
$GegNaam=switch($index1,"0","AC","BC");
$GegGetal:=maple("$f*($Geglijst[$index1])");
$Gegeven=decimal(2,$GegGetal);
$GevrNaam=switch($index2,"0","AH","BH","CH");
$GevrGetal:=maple("$f*($Gevrlijst[$index2])");
$antw=decimal(2,$GevrGetal);
```

Toelichting op bovenstaand *Algorithm*

Met het commando `triangle(T, [point(A, 0, 0), point(B, 2*$a, 0), point(C, $x, $y)])`: is de naam `T` van de driehoek gedefinieerd en daarmee ook de punten `A`, `B` en `C`.

Met het commando `altitude(hC, C, T, H)`: is de naam van de hoogtelijn `hC` gedefinieerd vanuit het hoekpunt `C` van de voorgedefinieerde driehoek `T` waarbij het hoogtepunt op de tegenovergelegen zijde meteen ook de naam `H` krijgt.

Met het commando `distance(C, H)` is bijvoorbeeld direct de afstand `CH` op te vragen.

TIP: In hoofdstuk 3 van de uitgave *Handleiding Maple 16* is een paragraaf over het geometry-pakket met nog meer voorbeelden te vinden.

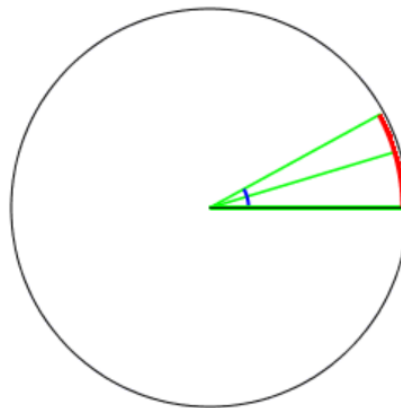
1.1.1.15 Animaties

In hoofdstuk 3 van de *Handleiding Maple 16* (zie (page 1)) is ook een paragraaf gewijd aan animaties: bewegende grafieken.

Een voorbeeld hiervan is het volgende:

Gegeven is de cirkel met straal (groen) $R = 2.1$.

De aangegeven hoek is $\alpha = 0.50$ (radialen),



TIP: Voer het antwoord in "rekenmachinetaal" in, dus haakjes, dakjes en sterretjes. of rond af op 2 decimalen.

a) De booglengte (rood) op de cirkel gelijk aan .

b) De oppervlakte van de bijbehorende cirkelsector (taartpunt) met deze gegevens is .

Figure 1.31: Animaties met een animatiegif in Maple gemaakt

Deze vraag bevat een geanimeerd plaatje dat in het *Algorithm* als volgt geprogrammeerd is:

```
$R=decimal(1,rand(0.5,5));
$A=decimal(1,rand(0.1,1.4));
$a=numfmt("#0.00",$A);
$boog=numfmt("#0.00",$R*$a);
$opp=numfmt("#0.00",1/2*$a*$R^2);
$p=plotmaple("
p1:=plot([t,0,t=0..1],color=green,thickness=3):
p2:=plots[polarplot]([1,phi,phi=0..
$a],scaling=constrained,thickness=5,color=red):
p2a:=plot([t,tan($a)*t,t=0..cos($a)],color=green):
p3:=plots[polarplot]([0.2,phi,phi=0..$a],thickness=2,color=blue):
```

```
p4:=plots[animate]( plot,
  [[t,tan(A)*t,t=0..cos(A)],color=green,scaling=constrained], A=0..
$a ,paraminfo=false):
plots[display]({p1,p2,p2a,p3,p4},tickmarks=[0,0]),
  plotoptions='height=250, width=250' ");
```

Zie voor meer informatie over animaties die met Maple te maken zijn in hoofdstuk 3 van de *Handleiding Maple16* (page 1).

TIP: Let hier ook eens op dat binnen de plotmaple-opdracht de voorbereide plots een naam krijgen met toekenning ($\coloneqq$) en zonder dollarteken. Binnen deze plotmaple-opdracht kunnen deze variabelen aangeroepen worden en met het Maple-commando `display` uit het plots-pakket tesamen in één figuur gevoegd worden. Buiten deze opdracht zijn de variabelen niet bekend.

TIP: kijk ook eens hoe het getal 0.9 in de tekst verschijnt met de opdracht *numfmt* (numeriek format voorschrijft), zodat het getal als 0.9 en niet als .9 in de tekst verschijnt, wat altijd veel verwarring veroorzaakt.

1.1.1.16 Lijnelementenveld

Het lijnelementenveld van een eerste orde differentiaalvergelijking is in beeld te brengen met behulp van het Maple-commando `dfieldplot` uit het DEtools-pakket. Zonder de optie `arrows=line` worden de lijnelementen met pijltjes afgebeeld.

In het *Algorithm* ziet u hoe at random een van de vijf DV's wordt gekozen waarvan een lijnelementenveldt wordt gemaakt.

```
$index=rint(5);
$kleur=switch(rint(6),"red","khaki","blue","turquoise","violet","sienna");
$dv1=maple("diff(y(x),x)=(y(x)+1)/x");
$dv2=maple("diff(y(x),x)=(x+y(x))/(x-2*y(x))");
$dv3=maple("diff(y(x),x)=(x-2*y(x))/(x+y(x))");
$dv4=maple("diff(y(x),x)=(2*x+1)/y(x)");
$dv5=maple("diff(y(x),x)=(x-y(x))/(2*x+y(x))");
$dv=switch($index,"$dv1","$dv2","$dv3","$dv4","$dv5");
$dvdisplay=maple("printf(MathML[ExportPresentation]($dv) )");
$plotdv=plotmaple("DEtools[dfieldplot]($dv,y(x),x=-5..5,y=-5..5,
color=$kleur, arrows=line, labels=[x,y]),
plotoptions='height=250, width=250' ");
```

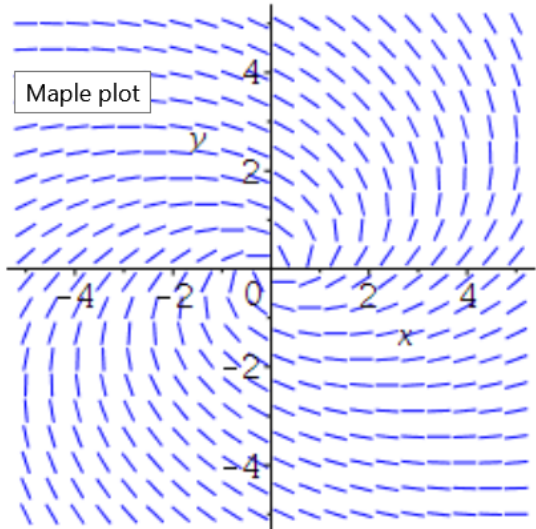

Variable	Value
index	1
kleur	blue
dv1	$\text{diff}(y(x),x) = (y(x)+1)/x$
dv2	$\text{diff}(y(x),x) = (x+y(x))/(x-2*y(x))$
dv3	$\text{diff}(y(x),x) = (x-2*y(x))/(x+y(x))$
dv4	$\text{diff}(y(x),x) = (2*x+1)/y(x)$
dv5	$\text{diff}(y(x),x) = (x-y(x))/(2*x+y(x))$
dvg	$\text{diff}(y(x),x) = (x+y(x))/(x-2*y(x))$
dvgdisplay	$\frac{d}{dx} y(x) = \frac{x+y(x)}{x-2y(x)}$
plotdvg	

Figure 1.32: Lijnelementenveld

TIP: Let op bij het definiëren van een differentiaalvergelijking dat hier bijvoorbeeld $y(x)$ dus echt als functie van x beschouwd wordt.

1.1.1.17 Complexplot

Een complex getal (of een lijst complexe getallen) in het complexe vlak tekenen gaat met het Maple-commando `complexplot` uit het `plotpakket`.

In de figuur ziet u hoe de student kan checken of zijn complexe getal de juiste plaats heeft in het complexe vlak.

Met een Maple-graded vraagtype kunt u bij de rubriek *Plotting Code* een Maple-opdracht programmeren zodat de P van plotting voor de student actief wordt.

De student ziet dan het volgende zoals in onderstaande figuur:

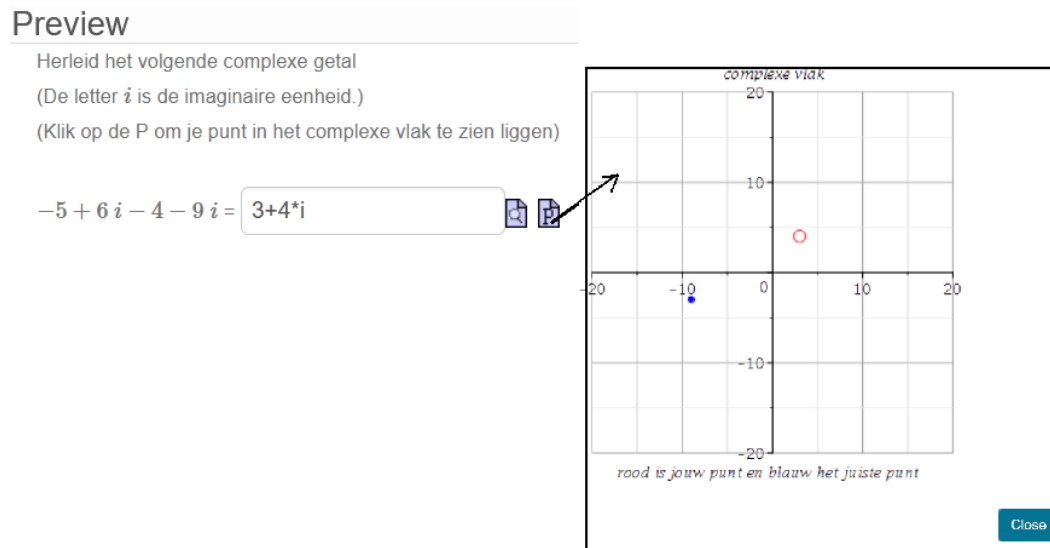


Figure 1.33: Complexe getallen in het complexe vlak

Algorithm voor de vraag:

```
$index=0;
$ij=switch($index, "i", "j");
$a1=switch(rint(2), range(-10, -2), range(2, 10));
$b1=switch(rint(2), range(-10, -2), range(2, 10));
$a2=switch(rint(2), range(-10, -2), range(2, 10));
$b2=switch(rint(2), range(-10, -2), range(2, 10));
$vraag="$a1+($b1)*I+($a2)+($b2)*I";
$vraagdispij=switch($index, "$a1+($b1)*i", "$a1+($b1)*j+($a2)+($b2)*j");
$vraagdisplay=maple("use InertForm:-NoSimpl in $vraagdispij: end:
printf(InertForm:-ToMathML(%))");
$antw=maple("$vraag");
$antwoord=maple("local I:=$ij: $antw");
$antwoorddisplay=maple("printf(MathML[ExportPresentation]
($antwoord))");
$plot=plotmaple("plots[complexplot]([$antw],
style = point, symbol = solidcircle, symbolsize = 15,color=blue,
title=`complexe vlak`,caption=`blauw is het juiste punt`,
gridlines=true,view= [-20..20,-20..20]),
plotoptions='height=300, width=300'");
```

TIP: Ten behoeve van verschillende notaties voor de imaginaire eenheid

$\$ij=switch(\$index, "i", "j");$ is hier in dit *Algorithm* met behulp van de variabele $\$index$ gekozen voor de letter i .

Voor de variabele $\$antwoord$ kunt u ook programmeren `maple("subs(I=$ij, $antw)");`

Grading Code:

In het Maple-graded antwoordveld hebt u de volgende *Grading Code* nodig om het antwoord van de student te matchen met het juiste antwoord:

```
evalb(($antwoord)-($RESPONSE)=0)
```

```
and evalb(StringTools[CountCharacterOccurrences]
("$RESPONSE", "$ij")<2);
```

TIP: Het betekent dat de imaginaire eenheid die hier met variabele \$ij aangeduid wordt, minder dan twee keer in het antwoord mag voorkomen.

Plotting Code:

De student kan met dit Maple Graded antwoordveld ook nog op de P klikken om een grafiek te zien als u het volgende in de *Plotting Code* invult:

```
resp:=subs($ij=I,$RESPONSE): antw:=$antw:
plots[display]([
plots[complexplot]([resp], style = point, symbol = circle,
symbolsize = 20,color=red),
plots[complexplot]([antw], style = point, symbol = solidcircle,
symbolsize = 10,color=blue)],
title=`complexe vlak`,caption=`rood is jouw punt en blauw het
juiste punt`,
gridlines=true, view=[-20..20,-20..20],scaling=constrained);
```

TIP:

Omdat de twee punten verschillende kleuren moeten hebben, maakt u twee plots die u met het commando `display` uit het plotpakket `over elkaar legt`.

Door de letter *i* die de student gebruikt te vervangen door de imaginaire eenheid *I* van Maple, wordt het studentenantwoord weer een echt complex getal en dat kan in het vlak getekend worden door middel van een punt.

Custom Previewing Code

Om te voorkomen dat de student gewoon de opgave overtipt en in de *Preview* ziet wat de vereenvoudiging is, programmeert u bij de *Custom Previewing Code* het volgende zodat het antwoord van de student niet automatisch vereenvoudigd wordt:

```
use InertForm:-NoSimpl in $RESPONSE: end: printf(InertForm:-
ToMathML(%));
```

1.1.1.18 Eigenplot

Het visualiseren van eigenvectoren van een 2×2 matrix kan met het Maple-commando `eigenplot` uit het LinearAlgebra-pakket van het Student-pakket.

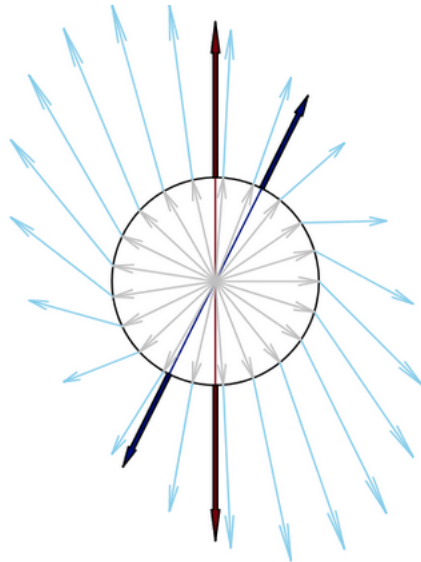


Figure 1.34: Eigenvectoren plot

In de figuur is de eigenplot in beeld gebracht van de matrix $\begin{bmatrix} 1 & 0 \\ -1 & \frac{3}{2} \end{bmatrix}$.

U ziet ongeveer 20 vectoren met lengte 1 (grijs) in alle mogelijke richtingen (eindigend op de eenheidscirkel).

De beelden van deze vectoren (blauw) onder deze matrix laat u beginnen aan de kop van ieder bijbehorend origineel.

Op die manier ziet u dat bijna elke vector iets gedraaid is ten opzichte van zijn origineel en soms iets langer of anders iets korter is geworden.

Er zijn maar twee richtingen waar de originelen en beelden in elkaars verlengde liggen (vet zwart).

De richting die behouden blijft is die van $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$ (blijft even lang) en de richting $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ (met factor $\frac{3}{2}$ gestrekt).

Deze twee (richtings)vectoren zijn de enige eigenvectoren van de gegeven matrix met respectievelijk eigenwaarden 1 en $\frac{3}{2}$.

De figuur krijgt u met de volgende opdracht in het *Algorithm*.

```
$plot=plotmaple("Student[LinearAlgebra][EigenPlot](Matrix([[1,0],
[-1,3/2]]),
showunitvectors, caption = ``, unitoptions = [color = grey, shape
= arrow],
imageoptions = [color = blue, shape = arrow]),
plotoptions='height=300, width=300');
```

TIP: Om te voorkomen dat de figuur een automatisch onderschrift krijgt, forceert u dat met het leeglaten van de caption met de optie `caption=```.

1.1.2 Labeled Images

Het is mogelijk om dynamische plaatjes te maken terwijl het plaatje zelf statisch is, maar de dynamiek ligt erin dat u tekst of getallen (of formules) in het plaatje kwijt kunt op een dynamische manier.

Het wil zeggen dat u bijvoorbeeld een plaatje neemt van een situatieschets waarin de getallen van de gegevens steeds veranderen.

U begint dan met een plaatje *zonder getallen* zoals hieronder afgebeeld, bijvoorbeeld een plaatje van een liggende balk met krachten en eenheden.

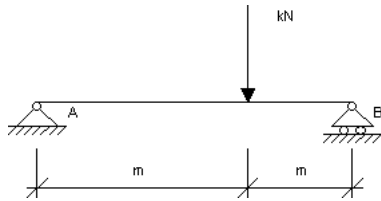


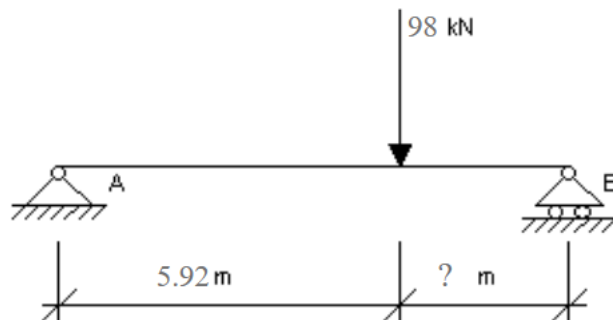
Figure 1.35: Situatieschets zonder gegevens

In de uiteindelijke vraag wilt u graag de gegevens, in dit geval de getallen op de lege plaatsen zien. Deze getallen moeten dan natuurlijk steeds variëren met de algoritmische variabelen zoals in de figuur hier onder te zien is.

Niet alleen getallen maar ook tekst in het plaatje kunt u natuurlijk ook op deze manier 'over het plaatje heen leggen'.

Gegeven is de onderstaande balk.

Gegeven is dat de verticale reactiekracht in punt B gelijk is aan 72.25 kN.



Gegevens in de figuur.

Hoe groot is dan de afstand op de balk waarbij het vraagteken staat?

Geef het antwoord (in meters) en rond af op 2 decimalen.

De afstand is m

Figure 1.36: Situatieschets met variabele gegevens

Dit lege plaatje kunt u opslaan in de *File Manager* van uw Class in het formaat .gif of .png (of .jpg).

TIP: Vergeet niet het plaatje van te voren op de **juiste afmetingen** te schalen en vervolgens uploaden in de *File Manager*.

De getallen (of tekst) worden er vervolgens als het ware 'overheen gelegd' door middel van het bepalen van de coördinaten waar de getallen moeten staan.

De getallen in het plaatje zijn de variabelen die u vooraf definieert in het *Algorithm* (zoals in de figuur hier onder) waar ook het correcte antwoord alvast wordt voorbereid.

```
$kracht=range(80,100);
$x=decimal(2,rand(3,6));
$y=decimal(2,rand(2,3));
$reactie=decimal(2,$kracht*$x/($x+$y));
```

Variable	Value	Range
kracht	94	80 - 100
x	5.27	3 - 6
y	2.11	2 - 3
reactie	67.12	

Figure 1.37: Algorithm van de belaste balk-reactiekracht

Vervolgens gaat u naar de tekst van de vraag. Om de code in te voegen, gaat u naar de broncode van de tekst van de vraag met behulp van de knop *Source*.

Het volgende script kan ingevoegd worden dus direct in de *broncode* van de tekst van de vraag.

```
<div class="labelledImage" style="height: 146px; width: 339px; float: none;">
  
  <div style="left: 200px; top: 140px;">?</div>
  <div style="left: 70px; top: 142px;">$x</div>
  <div style="left: 168px; top: 7px;">$kracht</div>
</div>
```

In de eerste regel staan de (**oorspronkelijke**) afmetingen van het plaatje: 339×146 (breedte × hoogte).

LET OP! Deze maten moeten beslist ingevuld worden met het oog op de coördinaten van de te plaatsen teksten of getallen.

In de tweede regel staat de url van het plaatje in een map van de *File Manager* in uw class.

Vervolgens moeten de variabelen een plaats krijgen in het plaatje. Daarvoor hebt u coördinaten nodig.

De linker bovenhoek van het plaatje is de oorsprong.

De eerste coördinaat geeft het aantal pixels naar rechts (vanaf de linker kant van het plaatje) en de tweede coördinaat geeft aan hoeveel pixels naar beneden vanaf de top (bovenkant van het plaatje). Het vraagteken is dus met coördinaten (200,140) in de figuur geplaatst.

U kunt ook attributen meegeven aan de tekst of de getallen (bijvoorbeeld **\$x**) of met aanpassen van lettergrootte, kleur of font zoals in het volgende iets uitgebreidere script:

```
<div class="labelledImage" style="height: 150px; width: 316px; float: none;">
```

```


<div style="left: 200px; top: 140px;">
  <span style="font-family:Times; font-size:16px; color:#ff0000;">?</span>
</div>
<div style="left: 70px; top: 142px; ">

  <span style="font-size:14px;">$x</span>

</div>
<div style="left: 168px; top: 7px; ">

  $kracht

</div>
</div>

```

TIP: Het is zelfs mogelijk om ergens in de figuur een formule te plaatsen met behulp van LaTeX.

```
<div style="left: 10px; top: 13px; ">\(a+\sqrt{b}\)</div>
```

Deze formule kan ook al voorbereid zijn in het *Algorithm* en kunt u hier in de broncode aanroepen met de naam van de variabele.

TIP: Om achter de juiste url van de afbeelding te komen, kunt u vooraf in de tekst van de vraag het plaatje invoegen met de knop (*Image*) en dan de url kopiëren. Meteen ziet u dan ook de oorspronkelijke maten van het plaatje. Vervolgens klikt u op *Cancel*.

TIP: Als u in de broncode van de tekst van de vraag het script hebt geplakt, en u vervolgens weer uit de broncode gaat, kunt u kijken hoe het er uitziet. Hou er rekening mee dat bij het aanpassen van vet of gekleurd, de plaats van de tekst iets anders kan uitvallen.

TIP: Hoe u in eerste instantie aan de coördinaten van de plek in het plaatje komt, is snel te achterhalen. U gokt eerst ongeveer. Maar u kunt ook het plaatje eerst even openen in bijvoorbeeld Paint en dan ziet u daar de coördinaten van plaatsen in het plaatje. Met de muis in het plaatje ergens gaan staan geeft onder in de context-bar van het programma Paint de coördinaten te zien in pixels.

TIP: Om de code te achterhalen van de attributen van de tekst zoals lettergrootte en kleur, maakt u deze tekst eerst aan in de vraag. Ga dan naar de broncode en kopieer deze code naar het script.

1.1.3 Free Body Diagram

Het vraagtype *Free Body Diagram* is vrij nieuw in Möbius. Een nadeel van dit vraagtype is dat het niet te randomiseren valt, maar daar staat tegenover dat u snel en gemakkelijk een hele serie van dit soort vragen kunt maken.

De student kan met de muis pijlen trekken vanuit een control point (B zie figuur). Deze pijlen kan de student als verschillende krachten aanmerken en labelen.

Het is ook mogelijk om in één figuur meer control points aan te merken en daarvanuit ook pijlen te definiëren.

De lengte van de pijl is bij een Free Body Diagram niet belangrijk, maar de richting wel.

1.1.3.1 Werking van de Free Body Diagram-vraag

Een blok met een massa van 10 kg bevindt zich op een wig met massa 21 kg zoals in onderstaande figuur. De statische wrijvingscoëfficiënt is 0.15 en het blok blijft op de helling van 30° stil liggen.

Teken het Free Body Diagram van alleen het blok.

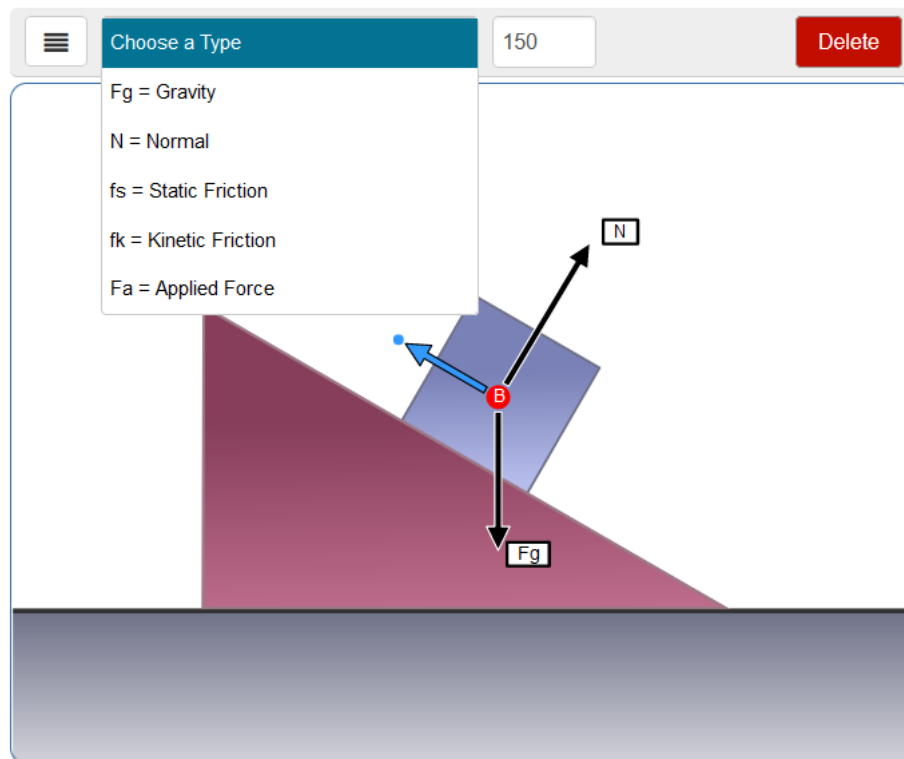


Figure 1.38: Pijlen in een Free Body Diagram

De student tekent de pijlen vanuit een control point met de muis. De eenmaal getekende pijl kan de student nog aanpassen wat richting en lengte betreft door het aanklikken van de pijl en het slepen met de muis.

De aangeklikte pijl krijgt een blauwe kleur en er verschijnt een handgreep (de blauwe punt bij de pijlpunt) om de pijl te verslepen. Eventueel kan de student met de *Delete*-knop de geselecteerde pijl weer weghalen.

Met een rolmenu (*Choose a Type*) kan de student de geselecteerde pijl labelen met behulp van een drop-down menu. En met het aantal graden kan de richting nog bijgesteld worden. De tolerantie in de richting is standaard 5° (maar kan aangepast worden).

De gradering van dit soort vragen is verder weinig flexibel. De soort kracht moet goed zijn (label), het totaal aantal krachten moet kloppen en de krachten moeten vanuit het juiste control point getrokken worden in de juiste richting met een tolerantie van 5° . Als één van de onderdelen niet juist is, dan is de beoordeling 0.

Meer control points

Meer control points is ook mogelijk, zoals in de volgende figuur waar de student met het menu de beschikking heeft over alle krachten die hij al getekend heeft. Aanvullingen voor de labeling kan de student alsnog maken en eventueel het aantal graden bijstellen.

In het ontwerp van de vraag zijn er veel mogelijkheden voor het labelen van het type kracht.

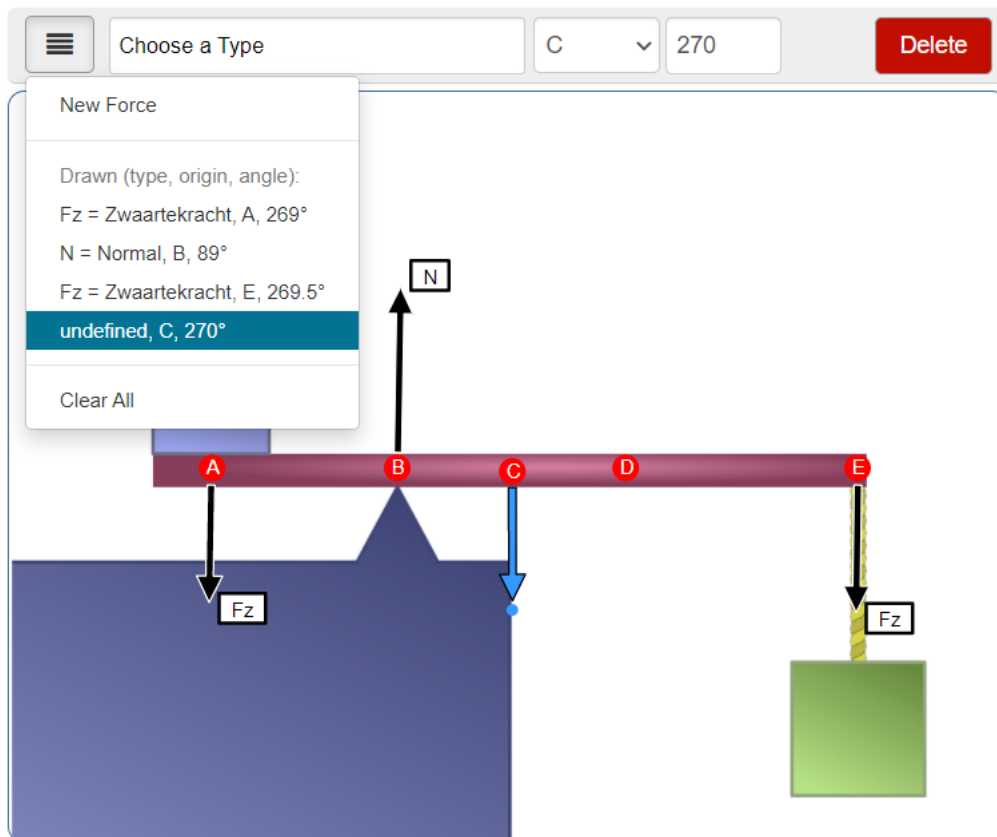


Figure 1.39: Ligger met box meer control points

1.1.3.2 Het ontwerpen van een Free Body Diagram-vraag

TIP: Voor het ontwerpen van een dergelijke vraag kunt u het beste met *Chrome* werken als browser. *Firefox* werkt voor dit vraagtype minder goed.

Geef een inleidende tekst en voeg een invulveld in en kies voor het vraagtype *Free Body Diagram*.

U ziet in de figuur het eerste deel van het dialoogscherm.

Edit Response Area

Question Feedback

Free Body Diagram

Weighting

Angle tolerance:

Background Image:

Show instructions

Select forces

- ☐ Fg = Gravity
- ☒ N = Normal
- ☒ T = Tension
- ☐ Fs = Spring Force
- ☒ fs = Static Friction
- ☒ fk = Kinetic Friction
- ☐ Generic = Generic Force
- ☒ Fz = Zwaartekracht

Define a new type of force

Abbr. Name

Figure 1.40: Editen van de Free Body Diagram Response Area 1

- In het dialoogvenster geeft u de *Weighting* voor dit invulveld (standaard = 1).
- Vervolgens vult u bij *Angle tolerance* in hoeveel de hoek van de pijl van de student mag afwijken (standaard = 5°) van de vastgestelde pijl.
- Vervolgens voegt u een plaatje in, liefst in png-formaat ter grootte van ongeveer 600 × 450 pixels.
 Als u een groter plaatje invoegt, dan wordt dat automatisch verschaald naar de genoemde maat met behoud van de ratio.
 Van te voren hebt u het plaatje reeds geupload in de *File Manager* en anders kan dat hier op de valreep ook nog.
 (Deel het plaatje wel in een van de mappen in.)
 Met de knop *Browse* komt u meteen in de *File Manager* van uw Class.
- Voor *Show instructions* vult u in *No*, want anders komen er Engelstalige instructies nogal storend in het oranje in beeld te staan voor de student.
 (Start drawing forces by clicking on a starting point and dragging your cursor to the desired area or make a selection below and click the add button.)

Eventueel vult u zelf instructies in om het applet met de pijlen te bedienen als het voor de student nieuw is.

- Dan ziet u een lijstje met mogelijke krachten staan die veel voorkomen in Free Body Diagrammen.
U kiest met de checkboxes de voorgedefinieerde krachten die u wilt aanbieden in het drop-down-menu van de student.
Eventueel kunt u zelf nog een paar extra krachten definiëren met een afkorting (Abbr.) en naam van de kracht (Name). Eventueel kunt u zo ook vertalingen maken. Dit nieuwe type kracht kunt u met de knop *Add force type* aan het default-lijstje toevoegen.
- Niet vergeten het lijstje te updaten met *Update force selection list below*.
- Naar onderen scrollen, komt u in het volgende scherm terecht waar u de control points invoegt zoveel als nodig:

control point label

List of control points:

A	Update Label	Delete
B	Update Label	Delete
C	Update Label	Delete
D	Update Label	Delete
E	Update Label	Delete

Add control points:

Figure 1.41: Editen van de Free Body Diagram Response Area 2

- U kiest nu één of meer control-punten waarbij u zelf een naam aan het punt kunt geven.
- Met de knop *Add* komt het punt ergens in de figuur te staan. Vervolgens kunt u het control point met de muis naar de juiste plek slepen.
- Bij het aanpassen van de naam van het control point niet vergeten op *Update Label* te klikken.

Vervolgens gaat u naar onder naar het plaatje

- In de figuur sleept u alle control points naar de juiste plaats en u vult de krachten in, niet door ze te tekenen met de muis (zoals de studenten zullen gaan doen), maar u kiest het type en de oorsprong (control point) en u vult de hoek in. Met *Draw* staat de kracht erin.

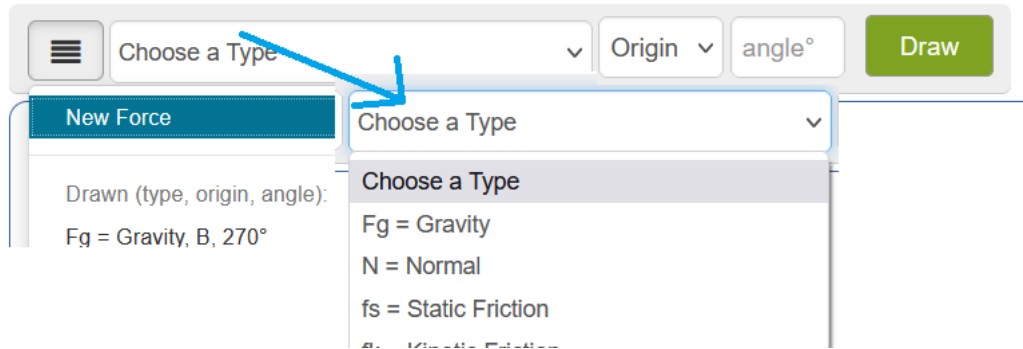


Figure 1.42: Definieren van krachten in Free Body Diagram

Uiteindelijk komt het plaatje er dan als volgt uit te zien, waarbij u een van de krachten kunt selecteren om te controleren en eventueel aan te passen.

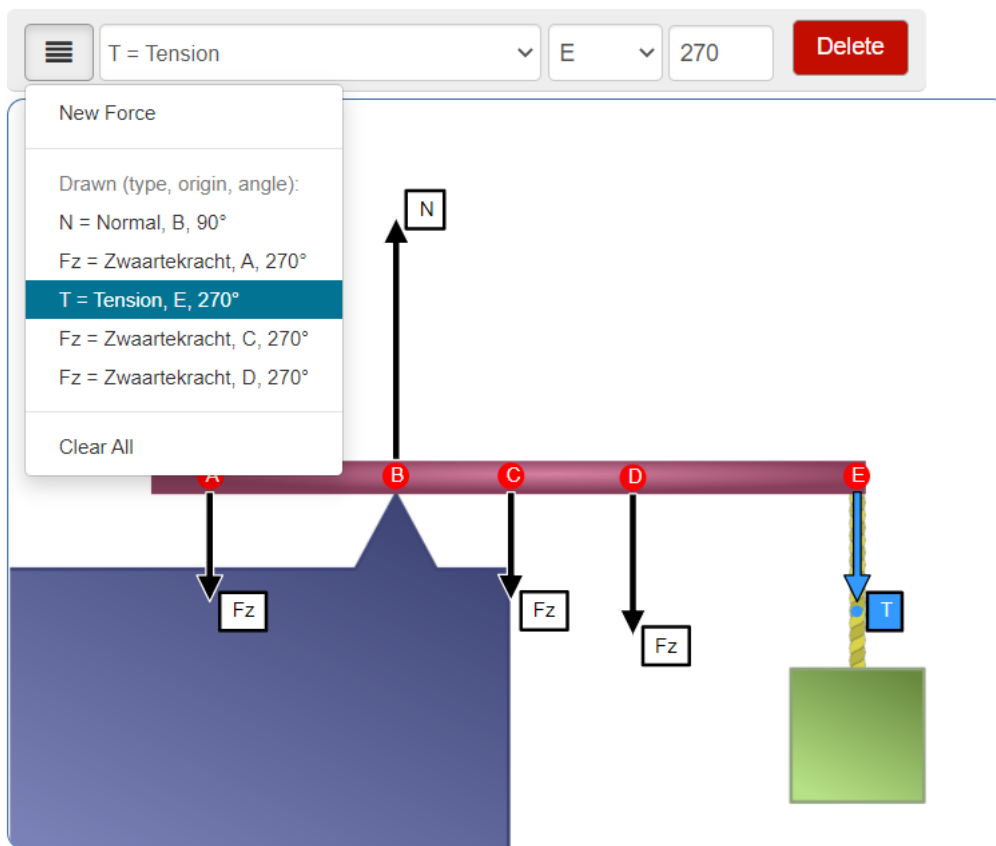


Figure 1.43: Editen van de Free Body Diagram Response Area 2

Ook via het menu bovenin het plaatje hebt u het overzicht en kunt u nog aanpassingen doen. Eventueel maakt u de pijl wat langer of korter met de muis en de blauwe pijlpunt door in het menu de juiste kracht te selecteren. (De lengte van de pijl is overigens niet van belang.)

TIP: Als u slechts één control-punt hebt aangebracht, ziet u de optie *Origin* niet staan.

Als u met de pijlen klaar bent, kunt u het *Response Area* afsluiten met *OK* en de vraag verder afmaken.

1.1.4 Sketch

Met het vraagtype *Sketch*, kan de student een grafiek tekenen. Het systeem doet de beoordeling of de grafiek van de student de juiste is met een bepaalde tolerantie. Er zijn slechts een paar soorten grafieken mogelijk.

Helaas is er geen randomisering mogelijk (ook niet door in de broncode te werken).

TIP: Deze sketchingvraag is bij uitstek geschikt om dwarskrachtenlijnen en momentenlijnen te tekenen. Maar er is meer mogelijk.

TIP: Er is veel informatie te vinden over dit vraagtype in de helpfunctie van Möbius met de zoekopdracht <scetch>.

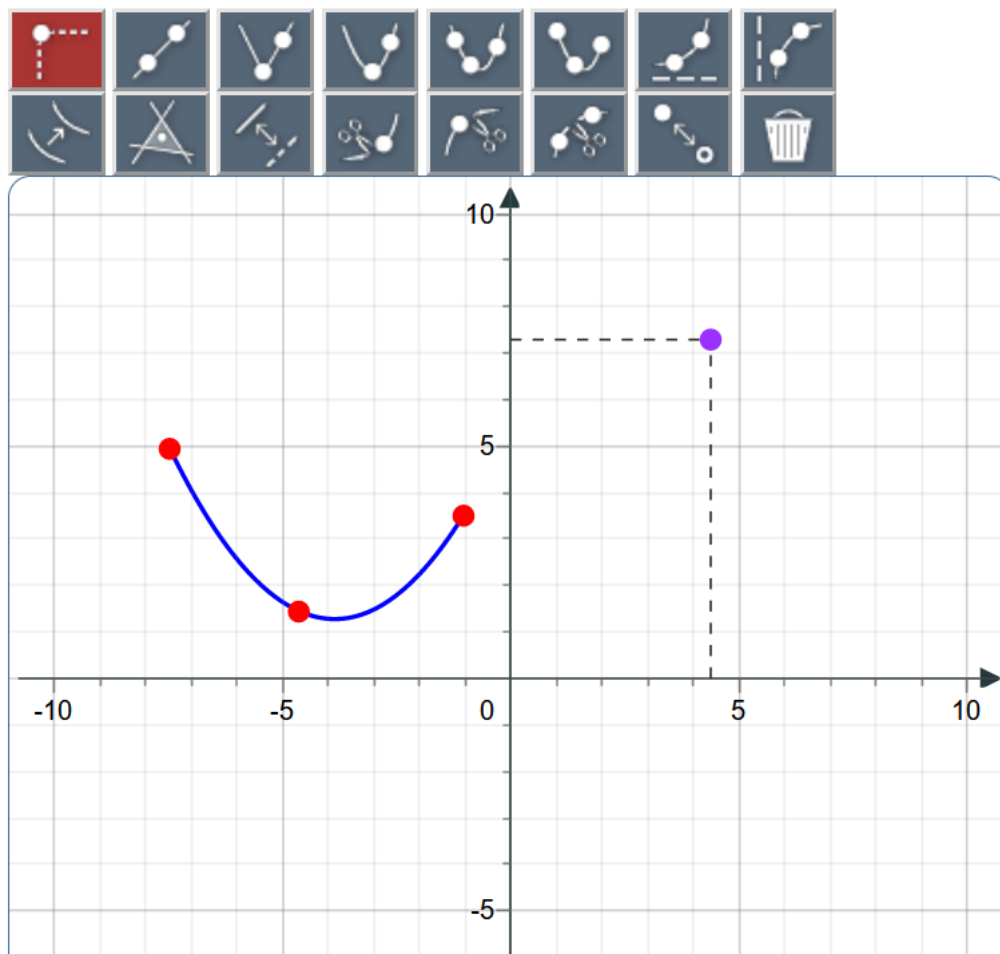


Figure 1.44: Sketchingvraag

1.1.4.1 Grafieken van functies schetsen

De functies die met deze tool geschetst kunnen worden zijn:

- 1) Dropline Point. Een punt met loodrechte afstanden tot de assen met stippellijnen.
- 2) Rechte lijn door twee punten (ook verticale lijn)

- 3) Absolutewaardefunctie met rechte lijnen met één knikpunt
- 4) Parabool met top en één ander punt
- 5) Parabool door drie punten
- 6) Paraboolsegment door drie punten
- 7) Exponentiële functie
- 8) Logaritmische functie
- 9) Een gepredefineerd stukje van een schets verplaatsen met de muis naar de juiste plaats.
- 10) Een gedeelte van het vlak selecteren dat aan één kant van een rechte lijn ligt of dat tussen twee rechte lijnen ligt.
Deze functie werkt alleen in combinatie met één of twee rechte lijnen.
- 11) Van een lijn een stippellijn maken
- 12) Het linker gedeelte van een grafiek weghalen
- 13) Het rechter gedeelte van een grafiek weghalen
- 14) Een tussengelegen deel van de grafiek weghalen
- 15) Een eindpunt definiëren als wel of niet open afsluiting.
- 16) Pullenmand om een onderdeel van de grafiek weghalen

De student ziet alleen de knoppen die nodig zijn om de schets te maken.

In onderstaande figuur moet de student een paraboolsegment tekenen door drie punten. De volgorde van aanklikken is niet belangrijk. Het linker en rechter punt vormen de eindpunten van het segment. Door te klikken op de prullenmand en vervolgens op de blauwe figuur, haalt u de schets van dit onderdeel weer weg.

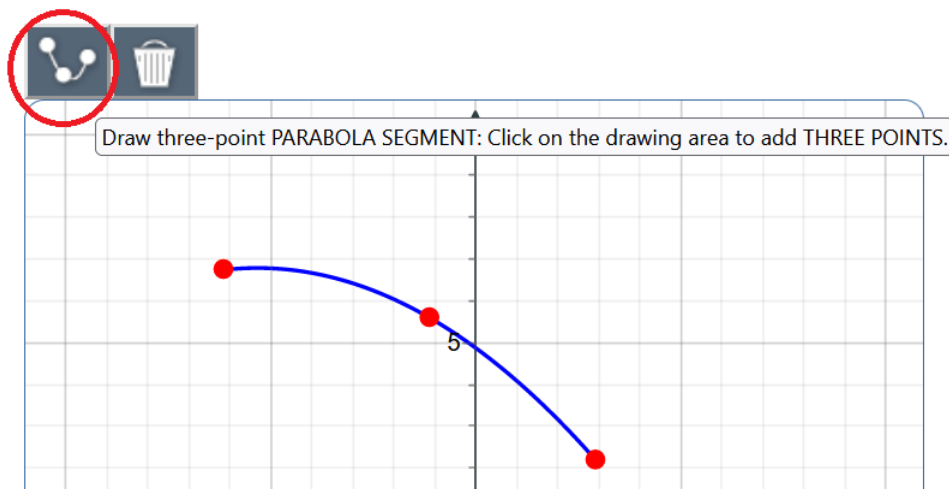


Figure 1.45: Sketchingvraag paraboolsegment

Alleen de knoppen die de student nodig heeft voor het maken van de schets, staan boven in beeld. In bovenstaande figuur is te zien hoe de student een paraboolsegment tekent. Hij klikt het pictogram van het paraboolsegment aan, en deze knop verschiet van kleur. Met mouse-over

is met de tekst "Draw three-point PARABOLA SEGMENT:" te zien dat er drie punten aangeklikt moeten worden. De punten verschijnen in het rood en meteen verschijnt ook het (blauwe) paraboolsegment. Als de student meer figuren dient te tekenen, krijgen deze allemaal automatisch verschillende kleuren. De bijbehorende knop zal dan beschikbaar zijn.

De student kan met de muis de rode punten verslepen en ziet dan meteen de coördinaten daarvan verschijnen. Daarna kan de knop van het paraboolsegment in het algemeen niet nogmaals aangeklikt worden, omdat in het ontwerp van de vraag wellicht aangegeven is dat de student slechts één parabool hoeft te tekenen. Als er meer van deze paraboolsegmenten nodig zijn, kan de student wel de knop nogmaals gebruiken.

Als de student een grafiek wil verwijderen, klikt hij eerst op de *Delete*-knop en vervolgens op de grafiek die hij wil weggooien.

De student kan een beperkt aantal grafieken tekenen:

TIP: Dus steeds eerst de knop aanklikken en daarna iets doen in de figuur.

	Dropline Point: Eén punt aanklikken
	Een rechte lijn: twee punten aanklikken
	Absolutewaardefunctie: eerst het knikpunt en daarna nog één ander punt aanklikken.
	Parabool: eerst de top en daarna nog één ander punt van de parabool.
	Parabool: drie punten aanklikken
	Paraboolsegment: drie punten aanklikken
	Exponentiële functie: eerst twee punten aanklikken die op de grafiek van de exponentiële functie liggen en daarna nog één punt aanklikken dat op de horizontale asymptoot ligt
	Logaritmische functie: eerst twee punten aanklikken die op de grafiek liggen van de logaritmische functie en daarna één punt van de verticale asymptoot
	Een figuur naar de juiste plaats verslepen of als achtergrond definiëren

1.1.4.1.1 Ontwerp van de vraag om een functie te schetsen

U begint met het openen van een *Response Area* en u kiest voor het vraagtype *Sketch*.

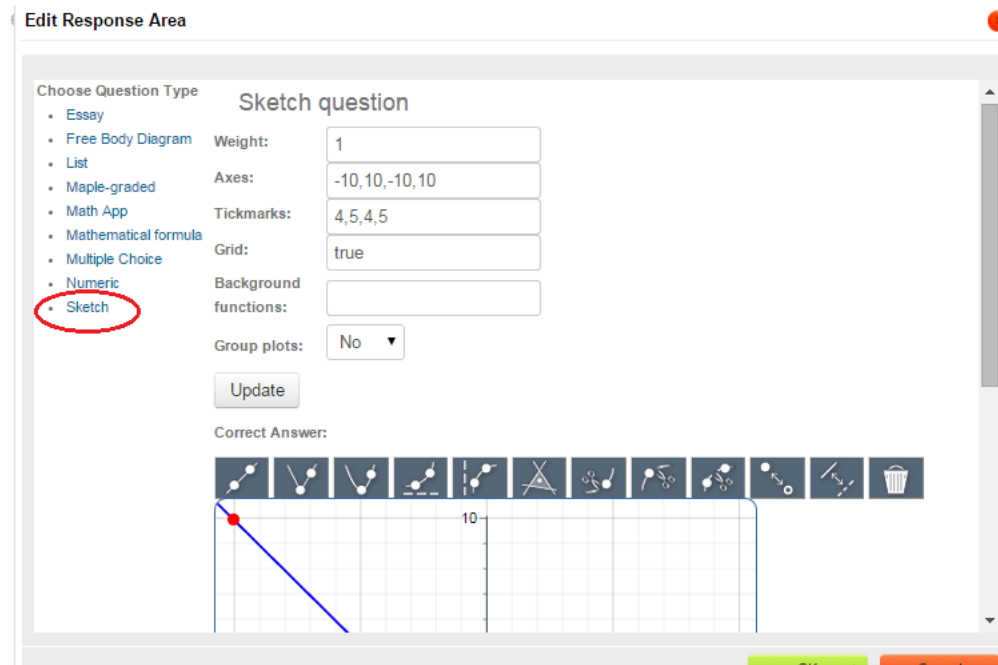


Figure 1.46: Sketchingvraag ontwerpen

U hebt in de ontwerpfase alle mogelijke knoppen tot uw beschikking. U vult vervolgens de *Weight* in (default = 1) en het bereik op de assen en de tickmarks. Bij de instellingen in de figuur hierboven resulteert *Axes* -10,10,-10,10 in een bereik op de *x*-as van $[-10, 10]$ en op de *y*-as eveneens. Bij *Tickmarks* betekent 4,5,4,5 dat de horizontale as in 4 gelijke delen wordt verdeeld en dat elk deel 5 tickmarks krijgt. Bij de verticale as idem. U kunt er ook voor kiezen de assen verschillend te behandelen.

Bij *Grid* kunt u er voor kiezen dat de lichtgrijs gekleurde gridlines niet (*false*) of wel (*true*) zichtbaar zijn. Echter er moet altijd wel iets ingevuld zijn bij *Axes*. Als u de rest dan leeg laat, krijgt u default gridlines.

Background functions laat u eerst even leeg. Het is namelijk mogelijk ook nog een achtergrondfunctie aan te bieden die de student steeds in beeld heeft. Zie daarvoor paragraaf (page 53).

Group plots zet u eerst op *No*. Zie paragraaf (page 50) voor meer informatie over deze knop.

Als u alle instellingen wilt accorderen, klik dan op de knop *Update*. U ziet dan in de figuur ook het juiste aantal gridlijnen en tickmarks verschijnen zoals u die hebt ingesteld. Eventueel aanpassen is mogelijk, maar dan ook weer op *Update* klikken.

Dan gaat u nu naar de grafiek en tekent, net zoals de student ook moet doen, een grafiek door eerst op een van de functieknoppen te klikken en dan met het vereiste aantal punten de grafiek vast te leggen. De student kan met andere punten volstaan om toch dezelfde grafiek te maken en dus een goede beoordeling te krijgen. U kunt met de rode punten slepen om deze te positioneren.

Alleen de knoppen die u gebruikt om de grafiek te maken, zijn voor de student zichtbaar. De knoppen die u niet gebruikt bij het maken van de grafiek zijn ook niet voor de student zichtbaar. Zie paragraaf (page 54) om eventueel nog iets met extra knoppen te doen door in de broncode te editen.

1.1.4.1.2 Meer grafieken in één figuur

Als u meer dan één grafiek wilt tekenen, bijvoorbeeld een lijn en twee parabolen, dan laat u de knop *Group plots* ook op *No* staan. U kunt dan meer grafieken in één figuur tekenen die

automatisch allemaal verschillende kleuren krijgen. Voor de student maakt het niet uit welke hij straks als eerste zal tekenen. Alle grafieken worden apart beoordeeld en de student krijgt deelpunten voor elke goede grafiek. De knoppen die u gebruikt om de grafiek te tekenen, komen automatisch beschikbaar voor de student. Als u bijvoorbeeld twee parabolen maakt, dan is de paraboolknop voor de student ook twee maal beschikbaar en niet meer om te voorkomen dat de student teveel maakt.

Wilt u echter dat als bijvoorbeeld minimaal één van de grafieken niet goed is dat dan de vraag geheel wordt beoordeeld met 0%, dan kunt u vóórdat u de grafieken begint te tekenen de instelling voor *Group plots* op *Yes* zetten en vervolgens niet vergeten op de knop *Update* klikken. Alle grafieken krijgen dan automatisch dezelfde kleur en horen bij elkaar. Eén of meer van deze grafieken fout resulteert in een 0% beoordeling.

1.1.4.2 Stuksgewijze grafieken schetsen

Er zijn nog meer knoppen mogelijk om bijvoorbeeld stuksgewijze grafieken te schetsen die aan elkaar gekoppeld kunnen worden.

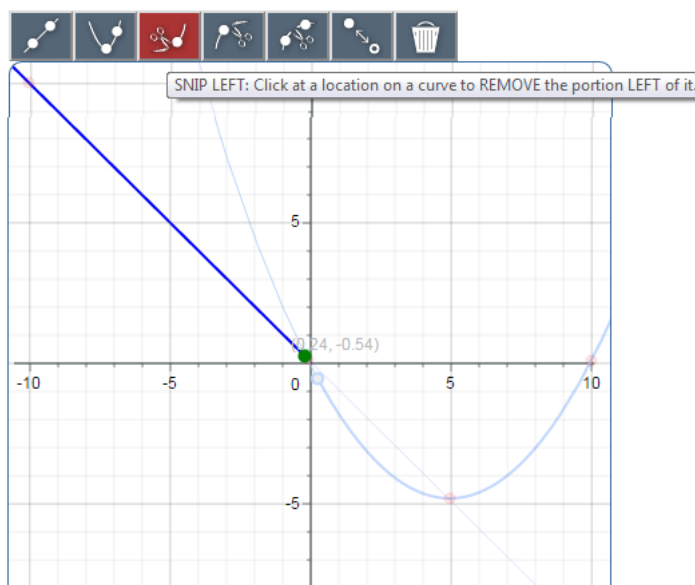


Figure 1.47: Sketchingvraag met stuksgewijze functies

De student trekt bijvoorbeeld een lijn door twee punten en laat het rechterstuk daarvan verdwijnen. Het eindpunt wordt dan automatisch groen gekleurd. Eventueel kan de student met een knopje (FILLED/HOLLOW) nog bepalen of dat eindpunt wel of niet meedoet met het linker stuk van de grafiek. Vervolgens maakt de student nog een andere grafiek erbij (bijvoorbeeld een parabool) en zorgt ervoor dat met SNIP LEFT het linker stuk verdwijnt. De student zorgt ervoor dat alleen de parabool van kleur verschiet om zeker te weten dat bij het bedoelde punt het linker stuk verdwijnt. Eventueel kan het groene eindpunt nog versleept worden om de twee grafieken te laten aansluiten.

TIP: dus steeds eerst de knop aanklikken en dan pas de actie in de figuur. Bij de *Delete*-knop is dat ook zo en als de student de subgrafiek wil weghalen moet hij op de *Delete*-knop klikken en dan twee maal de kromme aanklikken: eerst de subkromme, zodat die weer volledig wordt en daarna nogmaals op de kromme zelf. Bij SNIP BETWEEN zelfs drie maal op *Delete* klikken!

De student kan met de volgende knoppen gedeelten van grafieken weghalen en een sub-grafiek overhouden.



SNIP LEFT: Met deze knop kan het linker deel van de grafiek weggehaald worden.

	SNIP RIGHT: Met deze knop kan het rechter deel van de grafiek weggehaald worden.
	SNIP BETWEEN: Met deze knop kan door twee punten aan te klikken van een kromme een stuk van de grafiek ertussenuit gehaald worden.
	toggle FILLED/HOLLOW: om te switchen of het groene eindpunt wel of niet meetelt.

Deze knoppen kan de student in combinatie gebruiken met de functieknoppen en eventueel meer functies aan elkaar koppelen.

In het ontwerp van de vraag moet *vooraf* duidelijk zijn of u deze stukken één geheel laat vormen of ook afzonderlijk wilt beoordelen en dat moet dan eventueel ook aan de student gecommuniceerd worden. Overigens ziet de student in het resultaat aan het al of niet gelijk gekleurd zijn van de grafieken die hij tekent of ze bij elkaar horen of niet.

1.1.4.2.1 Stuksgewijze grafieken ontwerpen

De werkwijze is weer dezelfde als in de vorige paragraaf (*page 50*) bij het tekenen van meer grafieken in één figuur.

Als u wilt dat de stukken van de stuksgewijze grafiek bij elkaar horen, dan kiest u vóóordat u de grafieken tekent bij *Group plots* voor *Yes*. Maar u kunt de stukken natuurlijk ook afzonderlijk beoordelen als u bij *Group plots* kiest voor *No*.

TIP: Bij het gebruik van de SNIP-knoppen, verschiet de bedoelde grafiek van kleur en u kunt in het ontwerp een deel van de grafiek weghalen. Het groene eindpunt kunt u vervolgens verslepen naar de gewenste plek.

1.1.4.3 Lineaire ongelijkheden gebieden schetsen

Zoals in onderstaande figuur te zien is, kunt u ook lineaire ongelijkheden aan. Het is nogal beperkt, dat wil zeggen dat het dan over één of slechts *twee* lijnen moet gaan, zodat het gebied van het assenstelsel in tweeën (bij één lijn) of in vieren (bij twee lijnen) wordt gedeeld of in drieën als het om twee evenwijdige lijnen gaat.

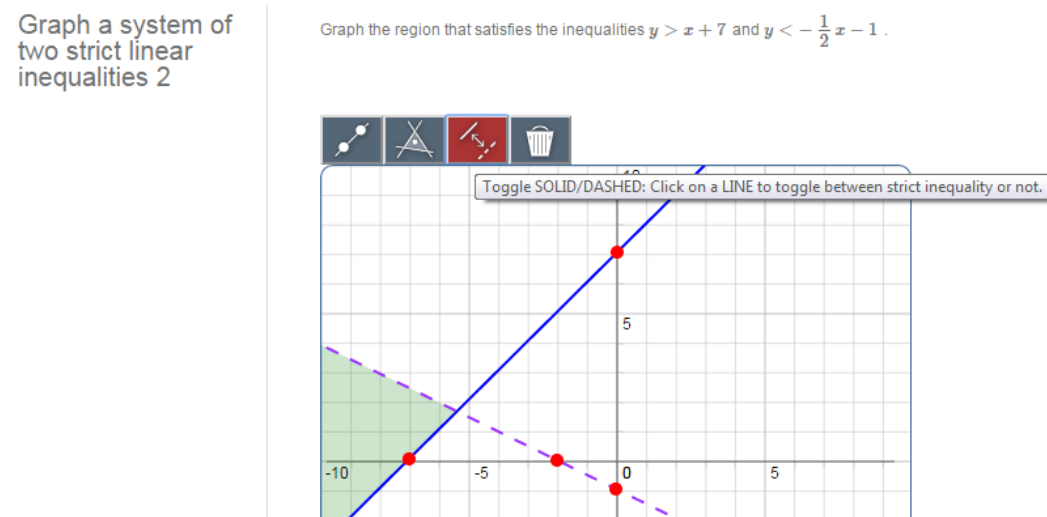


Figure 1.48: Lineaire ongelijkheid met sketching

Met deze actie is een drietal knoppen gemoeid:

	Een rechte lijn: twee punten aanklikken.
---	--

	Choose REGION: een punt in een gebied aanklikken.
	Wisselen tussen getrokken en gestreept

U maakt deze vraag weer op de gekende manier door in het ontwerp twee lijnen te tekenen en het juiste gebied aan te klikken en eventueel de lijnen al of niet gestippeld te maken met de Toggle SOLID/DASH-knop.

1.1.4.4 Achtergrondfunctie instellen

U kunt gemakkelijk een achtergrondfunctie instellen maar deze wordt dan dun en zwart weergegeven.

U vult dit in bij de rubriek *Background functions*: Deze figuur kan de student voortdurend zien.

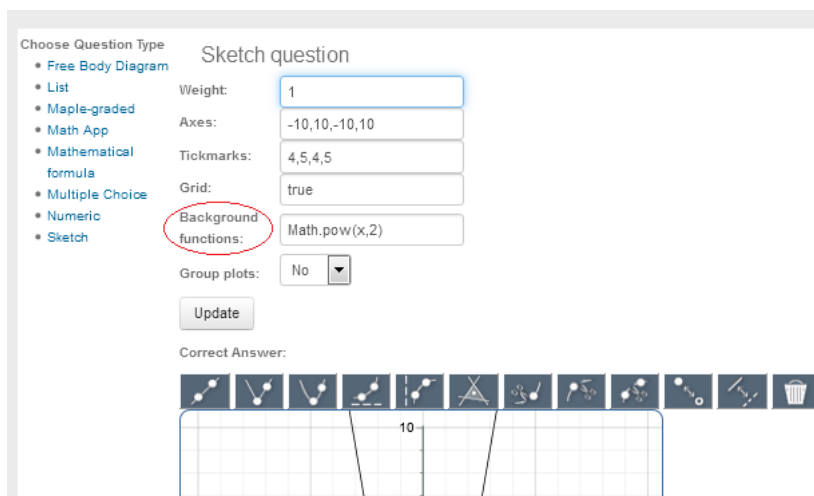


Figure 1.49: Achtergrondfunctie bij Sketching vraag

De achtergrondfunctie moet ingevuld worden als een functie genoteerd in javascript. (http://en.wikipedia.org/wiki/JavaScript_syntax#Math)

Enkele voorbeelden:

Math.cos(x)	Math.acos(x)	Math.log(x)	Math.exp(x)
Math.sin(x)	Math.asin(x)	Math.log10(x)	Math.sqrt(x)
Math.abs(x)	Math.pow(x,2)	simpele functies als 1/(x+1) en 4*x-2	Math.tan(x)

Geneste functies zijn ook mogelijk.

Math.sqrt(Math.sin(x)).

Binnen het argument wordt * en / wel verstaan dus $\sin(x^2)$ invoeren als Math.sin(x*x)

Eventueel ook 3*Math.pow(x,2)

TIP: Scheiden met semicolon voor meer achtergrondfuncties dus bijvoorbeeld:
Math.pow(x,2);Math.log(x)

TIP: Gestippelde achtergrondfunctie:
Math.pow(x,2):dashed;Math.log(x)

1.1.4.5 Aanpassingen in de broncode

Er is nog een aantal zaken die u niet in de ontwerp-dialoogbox van de vraag kwijt kunt, omdat bepaalde dingen automatisch gaan.

Zo is het aantal zichtbare knoppen in de vraag afhankelijk van het aantal dat u in het ontwerp van de vraag hebt gebruikt.

In paragraaf *text* (page 54) kunt u zien hoe u nog extra knoppen zichtbaar kunt maken. Ook is niet duidelijk hoe de tolerantie van de aangeklikte punten geregeld is. Daarover is in paragraaf *text* (page 54) nog meer informatie te vinden.

1.1.4.5.1 Tolerantie aanpassentext

De punten die u aanklikt in het ontwerp van de vraag kunt u later in de broncode eventueel nog wat beter positioneren, maar dat is niet echt nodig. De student klikt de punten ook niet precies aan. De tolerantie is meestal ruim voldoende. Het lijkt er echter op alsof de tolerantie iets te maken heeft met de plotview. Als u een vrij kleine range van de plotview genomen hebt, is de kans groot dat het moeilijker is om tot een goede grading te komen als de aangeklikte punten niet helemaal precies overeenkomen met die van het ontwerp. U kunt de tolerantie wat opschroeven door in de broncode van de vraag iets in de *Grading Code* aan te passen. Vul voor de tolerance een getal in tussen 0 en 1. Hoe groter dit getal des te groter de tolerantie. Een kwestie van uitproberen hoe u het hebben wilt vooral bij kleine ranges even opletten dus,

```
gradingCode=op(1,[Grading:-GradePlot($ANSWER, $RESPONSE,
$PLOTVIEW,tolerance=0.05)]);
```

1.1.4.5.2 Aantal zichtbare knoppentext

De knoppen die de student aangeboden krijgt bij de sketchingvraag zijn afhankelijk van de knoppen die u gebruikt hebt om het ontwerp van deze vraag te maken. In het voorbeeld van het stelsel lineaire ongelijkheden van paragraaf (page 52) zult u in de broncode vinden:

```
options="visible_buttons": "createline:2, togglesoliddashed, chooseregion, delete"@
```

Het wil zeggen dat de knop voor het trekken van een lijn twee keer gebruikt kan worden door de student, omdat u deze ook twee keer gebruikt hebt voor het maken van de vraag. Echter vindt u dat de student ook de mogelijkheid moet hebben om meer lijnen te trekken, dan maakt u in de broncode van de 2 een 3.

Andere mogelijkheden van combinatie van knoppen met de namen van de knoppen:

```
options="visible_buttons": "snipleft, snipright, snipbetween, togglefilled, createline:2, delete"@
options="visible_buttons": "createline:1, createabsolutevalue:1, createparabola:1,
createthreepointparabola:1, createexponential:1, createlogarithmic:1, delete"@
```

```
"visible_buttons": "createline:2, togglesoliddashed, chooseregion, delete"@
```

```
"visible_buttons": "createthreepointparabolaseg:1, delete"@
```

```
"visible_buttons": "snipleft, snipright, snipbetween, togglefilled, createline:3, delete"@
```

```
"visible_buttons": "snipleft, snipright, snipbetween, togglefilled, createline:2, createparabola:1,
delete"@
```

Zorg altijd ervoor dat u de delete-knop niet per ongeluk weghaalt in de broncode.

TIP: Als u alle knoppen voor de student in beeld wilt hebben en dat deze knoppen allemaal ook meermalen gebruikt mogen worden, dan laat u de regel leeg:

```
options=@
```

Tenzij er achtergrondfuncties zijn ingevuld dan ziet het er bijvoorbeeld als volgt uit:

```
options="background_functions": "Math.pow(x,2)"@
```

1.1.5 Organische chemie structuurformules

Structuurformules in de organische chemie in 3D die met de muis te draaien zijn, kunt u opnemen in uw item in Möbius.

U betreft dat van een website: <https://chemapps.stolaf.edu/jmol/jmol.php?> waar informatie te vinden is over de structuurformules. U hebt de officiële benaming van de stoffen nodig om de juiste structuur te krijgen die u wilt.

De internationale benaming van de structuur die u in de figuur hieronder ziet, is al in het *Algorithm* voor te bereiden met bijvoorbeeld een molecule dat u de variabele naam test geeft:

```
$test="1,2,2-trichloro-3,4-diiodocyclobutane";
```

Met mouse-over is te zien waar de verschillende kleuren voor staan als het om de afzonderlijke atomen gaat waaruit de structuur bestaat.

In de figuur ziet u dat bijvoorbeeld de groene atomen de Chloor-atomen (Cl) zijn en de paarse staan voor Jodium (I).

De grijze zijn de koolstof-atomen en de witte zijn de waterstof-atomen.

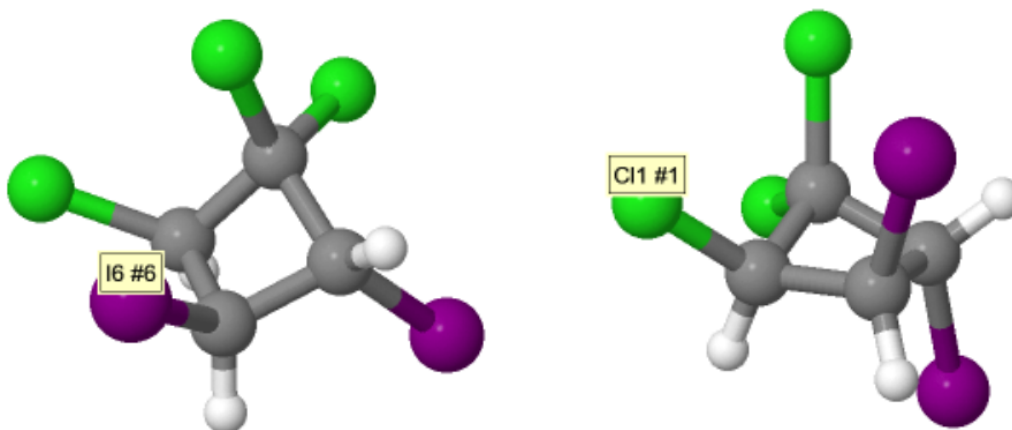


Figure 1.50: 3D structuurformule organische scheikunde

Maak de figuur in een iframe binnen een tabel

U maakt een dergelijke structuurformule in 3D het beste in een iframe binnen een tabel, waarbij u bijvoorbeeld in een van de kolommen van de tabel de begeleidende tekst plaatst. In het script van het iframe kunt u de naam van de molecule met een algoritmische variabele (\$test) aanroepen.

```
<p><iframe frameborder="no" height="400px" srcdoc="<script
  src='https://chemapps.stolaf.edu/jmol/jmol.php?model=
  $test&image3d&inline'></script>" width="100%"></iframe></p>
```

TIP: Wilt u liever de 2D-structuurformule zien, dan vervangt u in bovenstaande iframe-script alleen `image3d` in `image2d` en krijgt u de volgende figuur van dezelfde structuur waarin de atomen met hun symbool worden aangeduid: Cl voor chloor en I voor jodium met dezelfde kleuren als in de 3D-figuur.

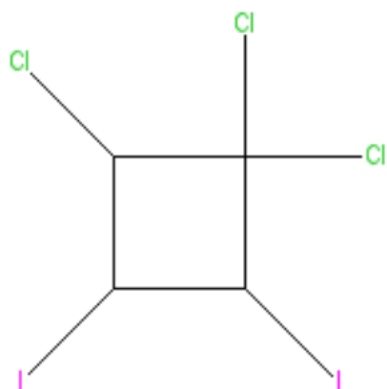


Figure 1.51: 2D structuurformule organische chemie

1.1.6 MathApp-questions

Een MathApp-antwoordveld maakt in de vraag van Möbius gebruik van een interactief Maple worksheet (*MathApp*) met embedded components.

Dit worksheet moet u eerst voorbereiden in het computeralgebrasysteem Maple dat u op uw computer moet hebben en vervolgens dit worksheet uploaden in de *Class File Manager*.

Het is mogelijk om de algoritmische variabelen in de vraag te laten corresponderen met de *MathApp*, zodat het interactieve worksheet ter plaatse aangepast kan worden met de geldende algoritmische variabelen van de vraag waarin het opgenomen is.

De student krijgt in de vraag het interactieve *Math App* te zien en hij kan de componenten daarvan, zoals sliders en punten in grafieken, aanklikken en dergelijke.

De stand waarin een (of meer) component van het *MathApp* door de student achtergelaten wordt, kan automatisch door Möbius beoordeeld worden door middel van de *Grading Code*.

In het Möbius item kiest u voor het *Math App* vraagtype.

De vraag komt er als volgt uit te zien en de student kan de slider instellen waarmee de figuur aangepast wordt.

Gegeven is de functie $f(x) = x^2 - 3x + 4$

Geef met behulp van de slider bij de grafiek aan welke de juiste grafiek is van deze functie.

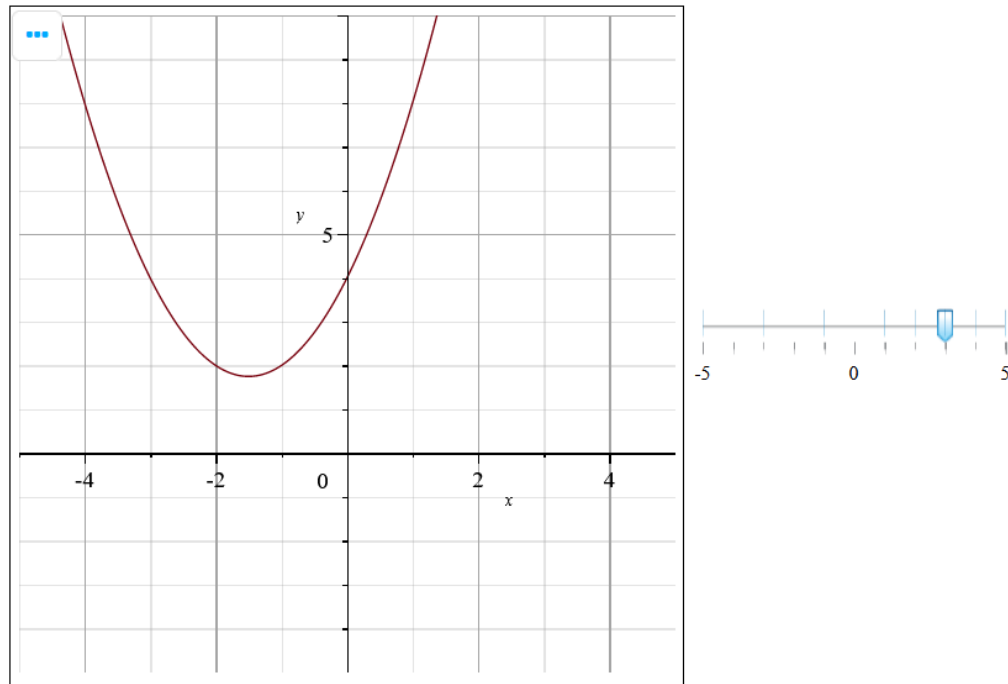


Figure 1.52: Math App in de vraag

In het *Algorithm* programmeert u het volgende waarmee u telkens een ander functievoorschrift krijgt.

De waarde van de variabele $\$a$ varieert en komt overeen met de waarde die de slider moet aannemen om de bijpassende grafiek te krijgen van het gegeven functievoorschrift:

```
$a=range(-5,5);
$fdisplay=maple("printf(MathML[ExportPresentation](f(x)=x^2+($a)*x+4))");
$p=plotmaple("plot(x^2+($a)*x+4,x=-5..5,y=-5..10,gridlines=true),
plotoptions='height=300, width=300'");
```

Het Maple-worksheet waarop deze Math App gebaseerd is, is al opgeslagen in de *File Manager*. In dat worksheet zijn twee componenten gedefinieerd: een figuur en een slider. De laatste heeft als naam *Slider0*. De naam van die component moet u kennen om de juiste *Grading Code* te kunnen maken voor de beoordeling van de student.

Aanmaken van een Math App-antwoordveld.

Met *Choose a Maple Worksheet file* kunt u het interactieve Maple Worksheet kiezen dat u eerst geupload is in de *File Manager* van de Class. En als u dat nog niet geupload heeft, kunt u dat hier op de valreep nog doen.

Question
Feedback

Math App:

Weighting

☒ **Worksheet File:** Choose a Maple Worksheet file
 

☐ **Worksheet URL:** Enter the URL of the MapleNet Worksheet

Worksheet Width:
☐ %
☒ px (200 - 1000) 

Initialization Parameters

Declare variables to be initialized by the Math App when the question is loaded. e.g.

foo=\$foo;

bar=\$bar;

☐ **No Grading** The Math App will be used for student exploration only.
☐ **Manual Grading** The Math App will be manually graded by the instructor.
☐ **Grading Algorithm in Math App** The Math App contains a grading procedure called Actions:-Grade
☒ **Grading Algorithm in Möbius** Use the following Maple code to grade the student response

Figure 1.53: Aanmaken van een Math App antwoordveld

U hoeft eerst nog niet iets met *Initialization Parameters* te doen. Dat komt als u wat meer geavanceerd hiermee gaat werken. Meestal zult u kiezen voor de optie *Grading Algorithm in Möbius*. Dan kunt u deze Math App in meer situaties inzetten.

Het betekent dat u hier nog wel een *Grading Code* moet programmeren om te beoordelen of de stand van de slider die de student achter laat de juiste is. U ziet alvast dat als de slider de waarde van \$a heeft, dat dat het juiste antwoord is.

Vervolgens kunt u kiezen voor het invoeren van het *Grading Algorithm in Möbius* (de *Grading Code*) om die in Möbius in te voeren.

- ☐ **No Grading** The Math App will be used for student exploration only.
- ☐ **Manual Grading** The Math App will be manually graded by the instructor.
- ☐ **Grading Algorithm in Math App** The Math App contains a grading procedure called Actions:-Grade
- ☒ **Grading Algorithm in Möbius** Use the following Maple code to grade the student response

Enter Maple code to grade the student response

```
if DocumentTools:-GetProperty(Slider0,value)=$a then
return 1;
else
return 0;
end if
```

Figure 1.54: Grading van een Math App vraag

Het betekent dus dat als de waarde van de Slider0 gelijk is aan de waarde van \$a, dan is de beoordeling goed en anders fout.

1.1.6.1 Voorbeeld van een eenvoudig Maple-Worksheet t.b.v de Math App

Het Maple-worksheet bevat de volgende componenten:

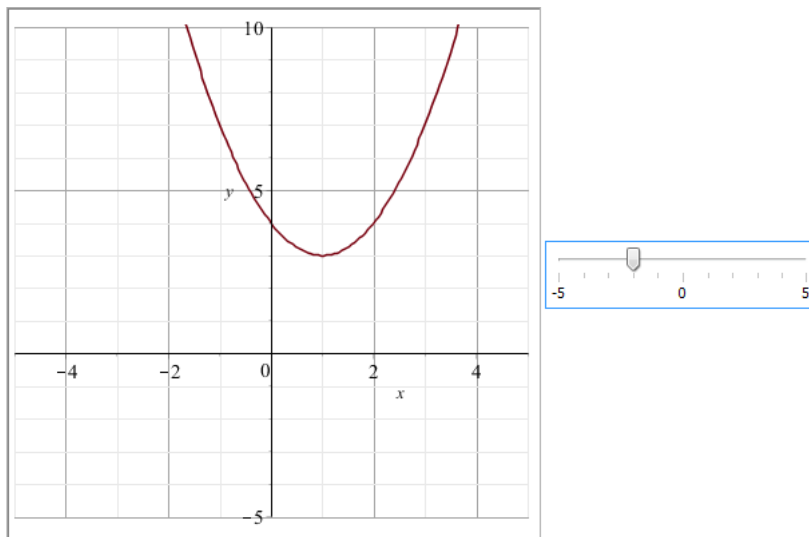


Figure 1.55: Math App eenvoudig Maple-Worksheet

Er is een plotcomponent (Plot0) en een slider (Slider0).

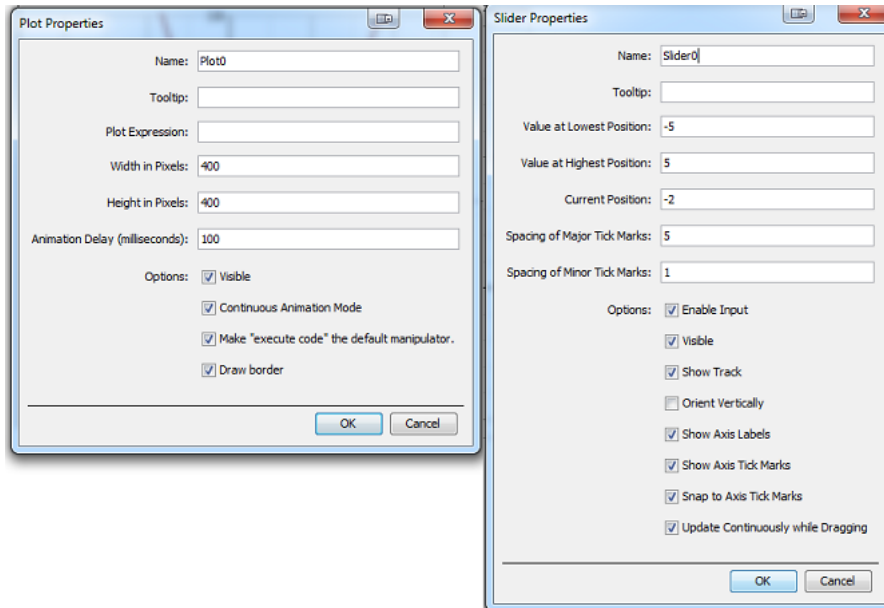


Figure 1.56: Eigenschappen van de componenten in het Maple Worksheet

Bij de plotcomponent hoeft geen plotcommando ingevoerd te worden, want daarvoor is de slider bedoeld.

Bij de slider voert u in bij *Action when value changes* de volgende regel toe:

```
use DocumentTools in
Do(%Plot0=plot(x^2+%Slider0*x+4, x=-5..5, y=-5..10,gridlines=true));
end use;
```

Het betekent dat als de Math App interactief ingezet wordt, dat bij het verschuiven van de slider de plot gegenereerd wordt, gerelateerd aan de stand van de slider.

De stand van de slider die de student achterlaat kan nu gebruikt worden om de *Grading Code* te maken:

```
if DocumentTools:-GetProperty(Slider0,value)=$a then
return 1;
else
return 0;
end if
```

U ziet dat hierin ook een algorithmische variabele gebruikt kan worden.

Het *Algorithm* is als volgt:

```
$a=range(-5,5);
$display=maple("printf(MathML[ExportPresentation])(f(x)=x^2+($a)*x+4)");
```

De vraag zou kunnen luiden:

Schuif de slider in de stand waarbij de grafiek overeenkomt met het functievoorschrift $\$display$.

De student moet de slider dan op de stand $\$a$ zetten om een 100% score te krijgen voor deze vraag.

TIP: Zorg ervoor dat u in het worksheet geen lege regels onder de componenten meer heeft staan anders krijgt u in de vraag in Möbius te veel witruimte.

1.1.6.2 Voorbeeld van een geavanceerd Math App Maple-Worksheet

Het Maple worksheet bevat de volgende componenten, opgemaakt in een tabel:

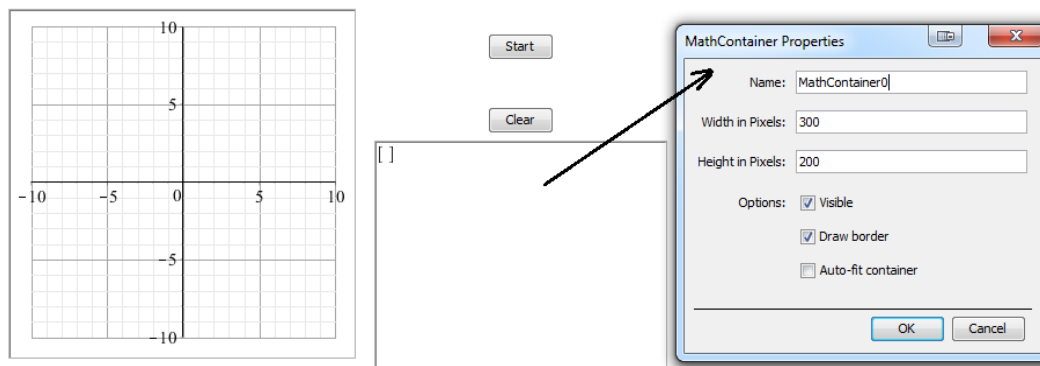


Figure 1.57: Math App worksheet in Maple

Een plotcomponent:

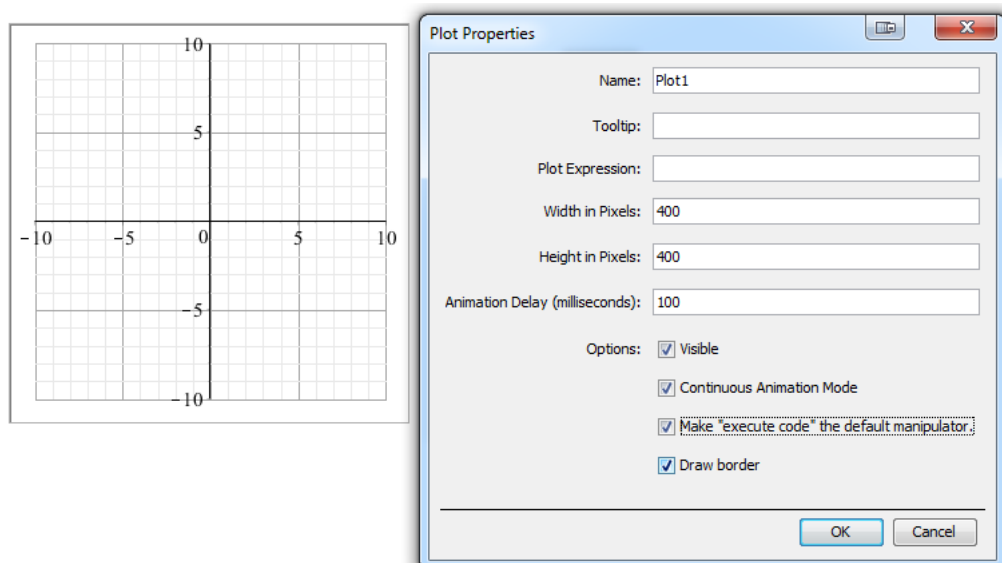


Figure 1.58: Math App Plotcomponent

In deze plot-component genaamd Plot1 zal de student straks punten moeten aanklikken. Bij de *Action When Clicked* kunt u volstaan met de procedure te noemen die in de *Startup-code* gedefinieerd is (`CurveFit:-Click()`);

Dan zijn er twee knoppen aangemaakt (Button0 en Button1).

Bij Button1 geven de de tekst "Start" aan deze knop en verbinden de actie `CurveFit:-Start()`; daaraan gekoppeld.

Bij Button0 geven we de tekst "Clear" aan deze knop en de actie `CurveFit:-Clear()`;

Verder nog een component *Mathcontainer* met de naam `MathContainer0` en die hoeft geen actie te hebben en wordt in de *Startup-code* onzichtbaar gemaakt. In deze *Mathcontainer* komen de punten te staan die de student aan gaat klikken en de inhoud van deze *Mathcontainer* kan dan ook gebruikt worden in Möbius bij de *Grading Code* om de grading van de vraag af te handelen.

Grading Code

Als de verzameling punten die de student invoert gelijk is aan de verzameling punten van het correcte antwoord:

```
Lijst := DocumentTools[GetProperty](MathContainer0, expression):  
Lijst1:=convert(Lijst,set): if $antw1 = Lijst1 then return 1 else  
return 0 end if;
```

Alternatieven voor de *Grading Code* zijn:

Als de doorsnede van de verzamelingen van het antwoord van de student en het correcte antwoord gelijk is aan de verzameling van het correcte antwoord:

```
Lijst := DocumentTools[GetProperty](MathContainer0, expression):  
Lijst1:=convert(Lijst,set):  
if $antw1 intersect Lijst1 = $antw1 then return 1 else return 0 end if;
```

Of als bijvoorbeeld alle tweede coördinaten van de ingevoerde punten gelijk moeten zijn aan -1.

```
Lijst := DocumentTools[GetProperty](MathContainer0, expression):  
if nops(Lijst) < 3 then return 0 else  
M:=convert(Lijst,Matrix): C:=LinearAlgebra[Column](M,2):  
n:=LinearAlgebra[Dimension](C):  
for i from 1 to n do if C[i] = -1 then return 1 else return 0 end if end do end  
if;
```

Of als de tweede coördinaat van alle punten gelijk moet zijn aan 0:

```
Lijst := DocumentTools[GetProperty](MathContainer0, expression):  
if nops(Lijst) < 3 then return 0 else  
M:=convert(Lijst,Matrix):  
C:=LinearAlgebra[Column](M,2):  
if LinearAlgebra[Norm](C,2)=0 then return 1 else return 0 end if  
end if;
```

Voor uitgebreide overhoring van een spline kunt u de volgende code gebruiken als u bijvoorbeeld ook nog wil eisen dat de helling positief is bijvoorbeeld of dat de grafiek hol is in zekere punten.

Het juiste lijstje met punten is bijvoorbeeld

```
$antw1:=maple("[[0,0],[2,-15],[4,-25],[7,21],[11,0]]");
```

Nog een grading code:

```
pointlist := DocumentTools[GetProperty](MathContainer0, expression):  
if  
nops(pointlist) < 3 then return 0 else  
spl:=CurveFitting[Spline](pointlist, x):  
erro:=0:  
grade:=0:  
for pt in $antw1 do  
erro:=erro+abs(pt[2]-eval(spl,x=pt[1]));  
end do;  
if erro<=2 then grade:=grade+0.5 end if;  
if (is(eval(diff(spl, x,x),x=5), positive)) and (is(eval(diff(spl, x,x),x=10),  
positive)) then grade:=grade+0.5;
```

```

else grade=0; end if;
return grade;
end if;

```

Als er bijvoorbeeld maar twee punten gecontroleerd hoeven te worden:

```

Lijst := DocumentTools[GetProperty](MathContainer0, expression):
Lijst1:=convert(Lijst,set):
lijn:=CurveFitting[PolynomialInterpolation](Lijst, x):
if evalb(diff(lijn,x)=$rc) then return 1 else return 0 end if;

```

1.1.6.2.1 Startup-code van de Math App

De startup-code vind u door in de menubalk bovenaan het icoontje met de twee wielmpjes aan te klikken.

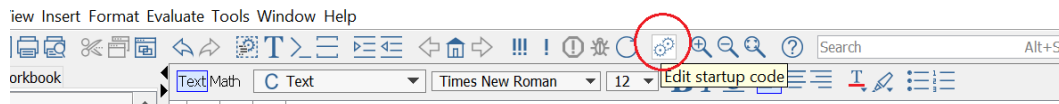


Figure 1.59: Knop in Maple voor de startup-code

Dan nu de *Startup-code* van dit *Maple Worksheet*, te beginnen met een module (Actions) om ervoor te zorgen dat de variabelen opgehaald worden vanuit het *Algorithm* van de vraag. Hier zijn dat 7 variabelen. Als u ook nog achtergrondgrafieken wilt hebben dan zijn er twee mogelijke functies en een mogelijke spline.

De labels, de view van de grafiek en het aantal aan te klikken punten door de student kan aangepast worden zodat deze MathApp in meer situaties te gebruiken is. Ook de nauwkeurigheid kan ingesteld worden met behulp van grid. Als de nauwkeurigheid bijvoorbeeld gelijk is aan 2, dan worden gridlines aangegeven en de ruimte tussen twee eenheden wordt dan in tweeën verdeeld. Default is de grid gelijk aan 1 zodat alleen roosterpunten kunnen worden aangeklikt. Steeds wordt een grafiek gemaakt van een vloeiende kromme (natural spline) door de aangeklikte punten. Aan de variabelen *listranges* (ten behoeve van de view van de grafiek), de labels en het aantal punten en de grid (gerelateerd aan het aantal gridlines en de nauwkeurigheid) worden defaultwaarden toegevoegd voor het geval deze variabelen verder niet gebruikt worden.

```

Actions := module()
  uses DT = DocumentTools;

  export InitParams;

  global symbolsizefix, CurveFit, BgC, BgF, BgG, BgPts, BgrngF, BgrngG,
    BgrngSpl, Vw, X, Y, Npt, Grx, Gry, fGrx, fGry, H, W, Fit, Clear, Click, Start;

  #initparams correspond with the algorithm of the question.
  #listranges will give the ranges of horizontal and vertical graph.
  #Now it is possible to use non integers for the ranges.
  #horlabel and vertlabel can also given in Maple syntax for example [m^2] etc for
  the labels of the axes.
  #npt gives the max amount of points there will be made a spline of the clicked
  points.
  #If the student gives more points then npt, then there will be no graph of the
  spline and he can press Clear.
  #There are now more possibilities to set the amount of gridlines and subticks
  (subgridlines) with gridx, factorgridx, gridy and factorgridy.

```

```

#These setting are connected to the tolerance.
#The student is able to click points on the gridlines. The point snips
automatically to a gridline nearest by.
#As an example: gridy=2 and factorgridy=10, and the range of the y-axis is 90.
#then the amount of horizontal gridlines is the y-range divided by 10 results in
9 gridlines.
#Each step of these horizontal gridline you can divide by 2 (= gridy) for
subticks (and subgridlines).
#this results in a tolerance of  $10/2 = 5$  units on the y-axis.
#so you can have less gridlines if you have big ranges.
#same story for the x-axis, independent of the grid on the y-axis.
#width and height will give the size of the plot.
#for the background there are different possibilities: two background functions
and one background spline.
#Bgcolor will set the color of the background graph.
#Bgf is the first background function with range Bgrangef which is default 0..0
to make it not visible.
#Same for the second background function Bgg and BgrangeG.
#Gives the list of points with BgPoints if you want a background spline with
range Bgrangespline.

InitParams := proc( {listranges:=[-10..10,-2..2],horlabel:"",vertlabel:"",
npt:=7,gridx:=1,factorgridx:=1,gridy:=1,factorgridy:=1,width:=400,height:=400,
Bgcolor:"gray",Bgf:=x^2,Bgg:=sin(x), BgPoints:[[ -4, 0], [ -2, 2], [ 0, 3], [ 2,
2], [ 4, 0]],
Bgrangef:=0..0,BgrangeG:=0..0,Bgrangespline:=0..0} )
global BgC,BgF,BgG,BgPts,BgrngF,BgrngG,BgrngSpl,Vw,X,Y,Npt,Grx,Gry,fGrx,fGry,H,W;

Vw:=listranges;
X:=horlabel;
Y:=vertlabel;
Npt:=npt;
Grx:=gridx;
Gry:=gridy;
fGrx:=factorgridx;
fGry:=factorgridy;
H:=height;
W:=width;
BgC:=Bgcolor;
BgF:=Bgf;
BgG:=Bgg;
BgPts:=BgPoints;
BgrngF:=Bgrangef;
BgrngG:=BgrangeG;
BgrngSpl:=Bgrangespline;

end proc;
end module:

# Deal with problem that 'symbolsize' is a relative measure

```

```

symbolsizefix := proc( n::integer, norm::integer := 400 )
    return round( n*norm/min( W,H) );
end proc; # symbolsizefix

CurveFit := module()
    uses DT = DocumentTools;
    export Fit, Clear, Click, Start;
    local DoPlot, Pts, C, Bgspl, x0, y0, Update;
    Actions[InitParams]();

Start := proc()
    Bgspl:= CurveFitting:-Spline( BgPts, x );
    Clear();
    DT:-Do(%Button1("visible")=false);
    DT:-Do(%Button0("visible")=true);
    DT:-SetProperty(Plot1, 'pixelHeight',H);
    DT:-SetProperty(Plot1, 'pixelWidth',W);
end proc;

Clear := proc()
    DT:-Do(%MathContainer0("visible")=false);
    Pts := [];
    DoPlot();
    DT:-Do(%MathContainer0=[]);
end proc; #Clear

Click := proc()

    x0 := DT:-Do(round(%Plot1( "clickx" )*Grx/fGrx)*fGrx/Grx );
    y0 := DT:-Do(round(%Plot1( "clicky" )*Gry/fGry)*fGry/Gry );

    Update();
end proc; #Click

Update := proc()
    Pts := [op(Pts), [x0, y0]];
    DT:-Do(%MathContainer0 = Pts);
    DT:-Do(%MathContainer0("visible")=false);
    Fit();
end proc: # Update

Fit := proc()
    if nops(Pts) < 3 or nops(Pts) > Npt then

```

```

        DoPlot();
        return NULL;
    end if;
    C := CurveFitting:-Spline( Pts, x );
    DoPlot(C);
end proc; #Fit

DoPlot := proc(C:=[ ])
    graph:=plots:-display( [
        plot( C,x=Vw[1], 'colour' = blue,'thickness'=3 ),plot( BgF,x=BgrngF,
        'colour' = BgC,'thickness'=1 ),
        plot( BgG,x=BgrngG, 'colour' =
        BgC,'thickness'=1 ),plot( Bgspl,x=BgrngSpl, 'colour' = BgC,'thickness'=1 ),
        plots:-pointplot(
            Pts,
            'style' = 'point',
            'symbol' = 'solidcircle',
            'symbolsize' = symbolsizefix( 15 ),
            'colour' = brown
        ) ], 'axis[1]'=[ 'gridlines'=[round(round(op(Vw[1]))[2]-(op(Vw[1]))
[1])/fGrx), 'color'=grey, 'subticks'=Grx-1
]], 'axis[2]'=[ 'gridlines'=[round(round(op(Vw[2]))[2]-(op(Vw[2]))[1])/fGry)
, 'color'=grey, 'subticks'=Gry-1]]
, 'view'=Vw, 'labels' = [typeset(X), typeset(Y)]
    );
    DT:-Do( %Plot1 = graph );
end proc; #DoPlot

end module; #CurveFit

# Initialize components
proc()
    uses DT = DocumentTools;
    Actions:-InitParams();
    DT:-Do(%Plot1=plot([0, 0], 'view' = Vw, 'color' = white, 'gridlines' =
    false, 'tickmarks'=[0,0], 'title'='First click the Start button`)); DT:-
    Do(%Button1("visible")=true);
    DT:-Do(%Button0("visible")=false);
end proc();

```

1.1.6.2.2 Initialiseren van parameters

Zoals gezegd begint de *Startup-code* van het Math App Worksheet met een module (Actions) om ervoor te zorgen dat de variabelen opgehaald zullen worden vanuit het *Algorithm* van de vraag.

Hier is dat een aantal variabelen voor de grafiek. De labels (horlabel en vertlabel), de view van de grafiek (litranges) en het aantal aan te klikken punten door de student (aantpt) kan aangepast worden en de grid die tevens de nauwkeurigheid verzorgt en tenslotte de afmetingen van de figuur met width en height, zodat deze MathApp in meer situaties te gebruiken is. Deze width en height kunnen niet boven de 1000 genomen worden.

Eventueel kunt u nog achtergrondgrafieken toevoegen in de vorm van een of twee functies en/of een spline met bijbehorende ranges zodat ook bijvoorbeeld twee grafieken aan elkaar geknoopt kunnen worden.

In het *Algorithm* van de vraag heeft u de betreffende variabelen ingevoerd:

```
$litranges=maple("[-3..3, -3..3]");
$horlabel="x";
$vertlabel="y";
$npt=7;
$gridx=1;
$gridy=1;
$antw1=maple("{[-3,0],[-2,0],[-1,-1],[0,0],[1,1],[2,0],[3,0]}");
```

In de *Math App Response Area* moeten deze variabelen dan ook gekoppeld worden met *Initialization Parameters*:

```
litranges=$litranges;
horlabel=$horlabel;
vertlabel=$vertlabel;
npt=$npt;
gridx=$gridx;
width=$width;
height=$height;
```

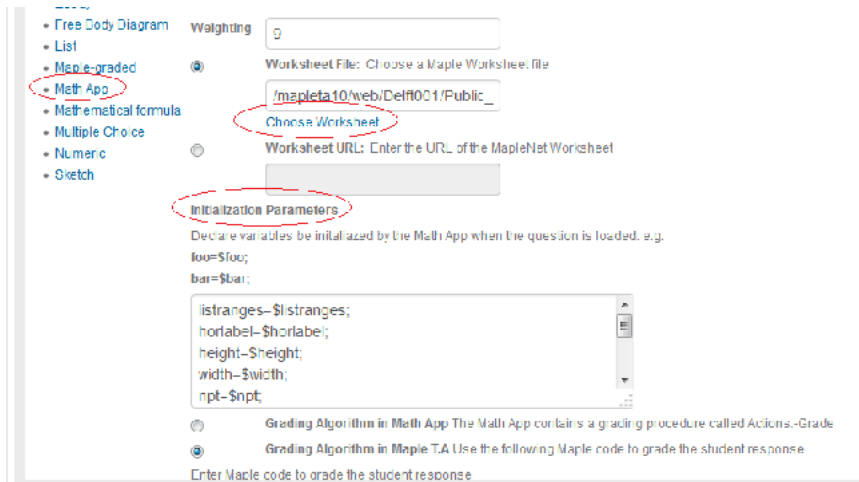


Figure 1.60: Algorithmische variabelen in de Math App

TIP: De *Startup-code* is zó geprogrammeerd dat als u bijvoorbeeld geen labels langs de assen van de grafiek wilt hebben, dan kunt u in het *Algorithm* volstaan met:

```
$horlabel="";
$vertlabel="";
```

of deze regels in het dialoogscherm van de math app weg te halen.

Ook is het mogelijk om eenheden in te voeren bij de labels. Gebruik dan wel Maple-syntax ten behoeve van de lay-out.

Zie bijvoorbeeld:

```
$horlabel=maple("x*[N/m] ");
$vertlabel=maple("y*[m/s^2]");
```

Als u de default ranges langs de assen wilt hebben, dan kunt u bijvoorbeeld in de *Initialization Parameters* de volgende regel weghalen:

```
listranges=$listranges;
```

In dat geval neemt het systeem de default ranges over.

Bij `grid=5` bijvoorbeeld, wordt de afstand tussen twee eenheden in het rooster in vijven verdeeld. De punten die de student aanklikt volgen het raster.

De waarden voor width en height zijn voor de afmetingen van de figuur.

TIP: Als u de MathApp in meer vraagstukken van één assignment wilt gebruiken waar misschien meer vraagstukken op één pagina komen te staan, ga dan de MathApp nogmaals uploaden onder een andere naam en gebruik in een ander vraagstuk het worksheet onder een andere naam.

TIP: Let ook op dat het worksheet geen overbodige witregels bevat, deze krijgt u namelijk dan ook in de vraag in Möbius als witregels.

1.1.7 SVG-figuren programmeren

Bent u bedreven in het programmeren van een SVG-figuur (Scalable Vector Graphics), dan zijn hier enkele tips.

Het voordeel is dat u gemakkelijk de variabelen voor de maatvoering van de figuur in het *Algorithm* kunt vastleggen waarmee de figuur dynamisch wordt en zich aanpast bij elke set van variabelen. Zelfs kleuren zijn te variëren.

Op <https://www.w3schools.com/graphics/> is een eenvoudige tutor te vinden om dit soort scripts te maken.

Oppervlakte van een rechthoek



In bovenstaande figuur staat een rechthoek met een hoogte van 50 mm en een breedte van 190 mm.

Geef de waarde van de oppervlakte

Figure 1.61: SGV-figuur van een rechthoek

Het Algorithm:

```
# instellingen voor de rechthoek
$width=range(50,200,10);
$height=range(50,200,10);
#kleur rgb
$R=$width;#afhankelijk van breedte
$G=$rint(255);# random
$B=$height;#afhankelijk van hoogte
$area=$width*$height;
```

Script van de figuur:

```
<svg height="$height" width="$width">
<rect height="$height" style="fill:rgb($R,$G,$B);stroke-width:1;stroke:rgb(0,0,0)"
width="$width"> </rect></svg>
```

TIP: Zet het script altijd in de vraag via de broncode en plaats het in een tabel!

Als u het script van de SVG-figuur niet in een tabel plaatst, komen er willekeurige witregels in de vraag te staan waar u ze niet wilt hebben!

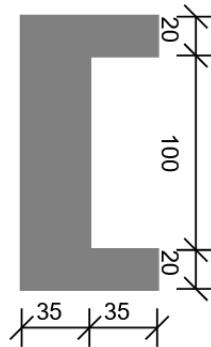
1.1.7.1 Berekeningen aan een profiel met dynamische figuur

Voor bijvoorbeeld de volgende vraag over een profiel met verschillende maatvoering:

U kunt voor de figuur de variabelen in het *Algorithm* gebruiken.

Hieronder volgt eerst het plaatje zoals dat in de vraag verschijnt.

Vervolgens het *Algorithm* van de vraag en ten slotte het script van het SVG-plaatje waarbij u gebruik kunt maken van de variabelen. Het plaatje wordt op deze manier dynamisch.



Bepaal de plaats van het zwaartepunt van dit profiel ten opzichte van het punt $[0, 0]$ linksonder.

$X =$ $Y =$

Figure 1.62: SVG-figuur

Algoritme:

```
$a=range(100,160,20);
$b=range(50,90,10);
$c=range(10,20,5);
$d=range(15,35,10);
#gegevens voor het plaatje
$height=$a+60;
$width=$b+60;
$orheight=$a-$c+30;
$fwidth=$b-$d+30;
$profileheight=$a+30;
$profilewidth=$b+30;
$brheight=$c+30;
$heightoffset=$a-$d-$c+30;
$mvheight=$a+50; #hoogte maatvoerlijn
$ahheight=$a+40; #hoogte aanhaallijn
$slheight1=$a+56; #bovenkant schuine lijn
$slheight2=$a+44; #onderkant schuine lijn
$xsl0=24; #X1 coördinaat schuine lijn 1
$xsl1=$fwidth-6; #X1 coördinaat schuine lijn 2
$xsl2=$profilewidth-6; #X1 coördinaat schuine lijn 3
$xsl3=36; #X2 coördinaat schuine lijn 1
$xsl4=$fwidth+6; #X2 coördinaat schuine lijn 2
$xsl5=$profilewidth+6; #X2 coördinaat schuine lijn 3
$xsl01=$width-16; #X coördinaat schuine lijn verticaal
$xsl02=$width-4; #X coördinaat schuine lijn verticaal
```

```

$ysl01=24; #Y1 coördinaat schuine lijn 1 verticaal
$ysl02=36; #Y2 coördinaat schuine lijn 1 verticaal
$ysl03=$brheight-6; #Y1 coördinaat schuine lijn 2 verticaal
$ysl04=$brheight+6; #Y2 coördinaat schuine lijn 2 verticaal
$ysl05=$orheight-6; #Y1 coördinaat schuine lijn 3 verticaal
$ysl06=$orheight+6; #Y2 coördinaat schuine lijn 3 verticaal
$ysl07=$profileheight-6; #Y1 coördinaat schuine lijn 4 verticaal
$ysl08=$profileheight+6; #Y2 coördinaat schuine lijn 4 verticaal
$mwwidth=$profilewidth+20; #X coördinaat verticale maatvoerlijn
$ahwidth=$b+40; #X coördinaat verticale aanhaallijn
$thx0=$((b-$d)/2)+20; #X coördinaat text 1 horizontaal
$thx1=$b-($d/2)+20; #X coördinaat text 2 horizontaal
$theight=$a+46; #hoogte-offset text breedte profiel
$bodywidth=$b-$d;
$bodyheight=$a-(2*$c);
$twidht=-$width+27; #breedte-offset text hoogte profiel
$widthoffset0=($c/2)+20; #hoogte-offset text 1 hoogte profiel
$widthoffset1=($a/2)+20; #hoogte-offset text 2 hoogte profiel
$widthoffset2=($a-($c/2))+20; #hoogte-offset text 3 hoogte profiel
#berekening
$A=($a*($b-$d))+(2*($d*$c));
$A1=($a*($b-$d));
$A2=($d*$c);
$X=((($A1*($b-$d)/2))+($A2*($b-($d/2))))/$A;
$Y=((($A1*($a/2))+($A2*($c/2))+($A2*($a-($c/2))))/$A;

```

Script voor het plaatje

```

<svg height="$height" width="$width"> <polygon points="30,30 30,$profileheight
$profilewidth,$profileheight $profilewidth,$orheight $fwidth,$orheight
$fwidth,$brheight $profilewidth,$brheight $profilewidth,30 30,30"
style="fill:grey;stroke:grey;stroke-width:2"></polygon> <horizontale
maatvoerlijn=" "></horizontale> <line style="stroke:black;stroke-width:1"
x1="30" x2="$profilewidth" y1="$mvheight" y2="$mvheight"></line>
<line style="stroke:black;stroke-width:1" x1="30" x2="30" y1="$height"
y2="$ahheight"></line> <line style="stroke:black;stroke-width:1"
x1="$fwidth" x2="$fwidth" y1="$height" y2="$ahheight"></line> <line
style="stroke:black;stroke-width:1" x1="$profilewidth" x2="$profilewidth"
y1="$height" y2="$ahheight"></line> <line style="stroke:black;stroke-width:1"
x1="$xsl0" x2="$xsl3" y1="$slheight1" y2="$slheight2"><text fill="black"
x="$thx0" y="$theight"> $bodywidth </text> <line style="stroke:black;stroke-
width:1" x1="$xsl1" x2="$xsl4" y1="$slheight1" y2="$slheight2"></line> <text
fill="black" x="$thx1" y="$theight"> $d </text> <line style="stroke:black;stroke-
width:1" x1="$xsl2" x2="$xsl5" y1="$slheight1" y2="$slheight2"></line>
<verticale maatvoerlijn=" "></verticale> <line style="stroke:black;stroke-
width:1" x1="$mvwidth" x2="$mvwidth" y1="30" y2="$profileheight"></line> <line
style="stroke:black;stroke-width:1" x1="$ahwidth" x2="$width" y1="30" y2="30"></
line> <line style="stroke:black;stroke-width:1" x1="$ahwidth" x2="$width"
y1="$brheight" y2="$brheight"></line> <line style="stroke:black;stroke-
width:1" x1="$ahwidth" x2="$width" y1="$orheight" y2="$orheight"></line>
<line style="stroke:black;stroke-width:1" x1="$ahwidth" x2="$width"
y1="$profileheight" y2="$profileheight"></line> <line style="stroke:black;stroke-
width:1" x1="$xsl01" x2="$xsl02" y1="$ysl01" y2="$ysl02"></line> <text
fill="black" transform="rotate(90 0,0)" x="$widthoffset0" y="$twidht"> $c
</text> <line style="stroke:black;stroke-width:1" x1="$xsl01" x2="$xsl02"
y1="$ysl03" y2="$ysl04"></line> <text fill="black" transform="rotate(90
0,0)" x="$widthoffset1" y="$twidht"> $bodyheight </text> <line
style="stroke:black;stroke-width:1" x1="$xsl01" x2="$xsl02" y1="$ysl05"
y2="$ysl06"></line> <text fill="black" transform="rotate(90 0,0)"
x="$widthoffset2" y="$twidht"> $c </text> <line style="stroke:black;stroke-
width:1" x1="$xsl01" x2="$xsl02" y1="$ysl07" y2="$ysl08"></line> </svg>

```

1.2 Werken met Maple Repositories

Het is mogelijk om in speciale gevallen een zelfgemaakte Maple routine in te zetten bij onderdelen van de vraag.

Soms moet Maple iets ingewikkelds voor u doen ten behoeve van een vraag en dan is het handig om deze actie in een zelfgeschreven procedure onder te brengen en vervolgens deze procedure aan te roepen in het *Algorithm* of de *Grading Code* van de vraag.

1.2.1 De module met procedures in Maple voorbereiden

Hier volgt een voorbeeld van een tabel die in Excel gemaakt is en in Maple ingelezen moet worden waarna Maple iets moet doen met deze tabel.

Het kan handig zijn om een aparte module met procedures te schrijven die op deze tabel gebaseerd zijn zonder die tabel later te communiceren in de vraag zelf of in het *Algorithm* van de vraag. U volstaat dan gewoon met het aanroepen van de procedure.

U opent nu eerst in het computeralgebrasysteem Maple een Worksheet en sla het op met de naam "MatrixInlezen".

In dit Worksheet gaat u de module met procedures maken.

In dit voorbeeld wordt er een Excel-bestand binnengehaald (hier als voorbeeld een tabel bestaande uit 41 rijen en twee kolommen).

Er kan een element of een rij of een kolom van deze matrix worden gegenereerd door het schrijven van een module met procedures.

Onderstaande module (die u dezelfde naam geeft als het Worksheet: "MatrixInlezen") bevat drie procedures (genaamd "RijM" en "KolomM" en "ElementM").

Eventueel kunt u deze module aanvullen met meer procedures als er meer van dergelijke procedures nodig zijn.

Vooraf haalt u een bestaand Excel-bestand binnen in uw Maple Worksheet. Het kan zowel een .xls als .xlsx Excel-bestand zijn, dat maakt niet uit.

Zorg dat dit Maple Worksheet in dezelfde directory staat opgeslagen als het binnen te halen Excel-bestand, dan is het gemakkelijk te vinden.

Voor de module die u nu gaat maken kiest u dezelfde naam als het Worksheet (is niet beslist noodzakelijk maar wel handig).

Met de volgende opdracht in uw Maple Worksheet, krijgt u een dialoogscherf te zien voor het importeren van een tabel.

```
> ExcelTools[Import]();
```

U navigeert naar de plaats waar uw Excelbestand staat:

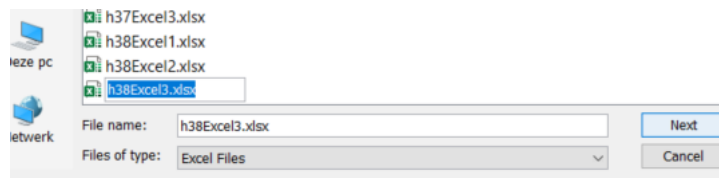


Figure 1.63: Navigeren naar het Excelbestand

Met *Next* krijgt u het volgende dialoogscherf waarbij u het werkblad en het celbereik van uw Excelbestand opgeeft.

Voor een rechthoekige matrix geeft u de cel linksboven tot en met de cel rechtsonder.

Hier is dat bijvoorbeeld B4 : C44 voor een matrix van 41 rijen en 2 kolommen.

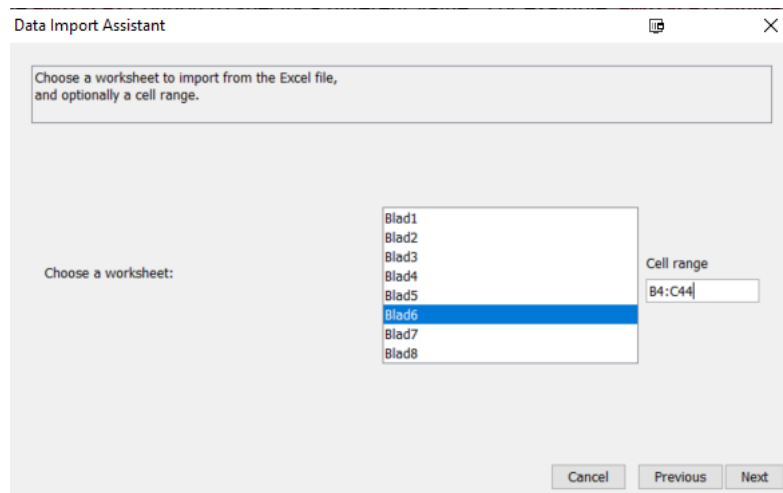


Figure 1.64: Werkblad en Celbereik opgeven

Met klikken op *Next* krijgt u het volgende dialoogscherm te zien waarbij u kiest voor *Rectangular*:

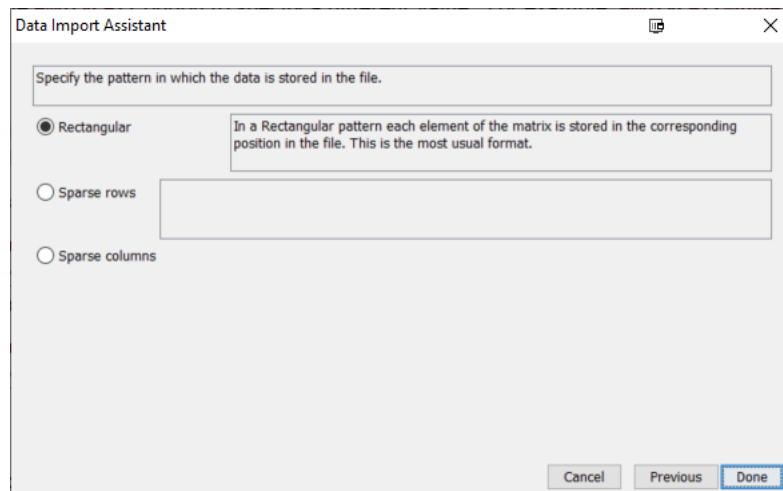


Figure 1.65: Rechthoekig bereik

U hebt nu een 41×2 matrix gekregen in de uitvoer van uw Maple Worskheet. Deze uitvoer kunt u kopiëren en straks plakken in de te maken procedure.

Nu begint het maken van de module waarbij het definiëren van de matrix M binnen de procedure gedaan kan worden door bovenstaande output dus te plakken en te definiëren als de Matrix M binnen de module hieronder.

```
> # Definieer de module
restart;MatrixInlezen:=module()
  option package;
  # Definieren dat de gedeclareerde matrix binnen de module te gebruiken
  is
    local M:

    #Procedures beschikbaar ook buiten de module
    export RijM, KolomM, ElementM;

    # Definieren van de matrix M door met copy en paste te werken na het
    eerder inlezen van de matrix
```

```

M:=Matrix(41, 2, [[0.,18.0],[10.0,18.0],[20.0,18.0],[30.0,18.0],
[40.0,17.9],[50.0,17.9],[60.0,17.8],[70.0,17.8],[80.0,17.7],
[90.0,17.6],[100.0,17.5],[110.0,17.4],[120.0,17.3],[130.0,17.2],
[140.0,17.0],[150.0,16.9],[160.0,16.7],[170.0,16.6],[180.0,16.4],
[190.0,16.2],[200.0,16.0],[210.0,15.8],[220.0,15.6],[230.0,15.4],
[240.0,15.1],[250.0,14.9],[260.0,14.6],[270.0,14.4],[280.0,14.1],
[290.0,13.8],[300.0,13.5],[310.0,13.2],[320.0,12.9],[330.0,12.6],
[340.0,12.2],[350.0,11.9],[360.0,11.5],[370.0,11.2],[380.0,10.8],
[390.0,10.4],[400.0,10.0]]);

#Procedure definitions
RijM:=proc(i)
  local R:
  R:=LinearAlgebra[Row](M, [i]);
  return R;
end proc: #RijM

KolomM:=proc(j)
  local K:
  K:=LinearAlgebra[Column](M, [j]);
  return K;
end proc: #KolomM

ElementM:=proc(i,j)
  local E:
  E:=M[i,j];
  return E;
end proc: #ElementM

end module:#MatrixInlezen

> #Check welke procedures de module MatrixInlezen dus nu bevat
with(MatrixInlezen);

[ElementM, KolomM, RijM] (1.1)

> #Try-out van de procedures
#tweede kolom van de matrix
MatrixInlezen[KolomM](2);

[ 18.0
  18.0
  18.0
  18.0
  17.9
  17.9
  17.8
  17.8
  17.7
  17.6
  ⋮ ] (1.2)
41 element Vector[column]

> # 8ste rij van de matrix
MatrixInlezen[RijM](8);

[ 70.0 17.8 ] (1.3)

```

```
> # Het element met index [8,1]
  ElementM(8,1);
```

70.0

(1.4)

Als u tevreden bent over de werking van de procedures, exporteer (save) de module dan nu naar een library-bestand waarmee er in feite **twee bestanden** aangemaakt worden: een .lib en een .ind file. Deze twee bestanden moet u nu in dezelfde directory zien staan op uw computer als waar u ook het Maple Worksheet hebt opgeslagen.

De aangemaakte module met procedures slaat u als volgt op:

```
> LibraryTools[Save](MatrixInlezen, "MatrixInlezen.lib");
```

Deze twee bestanden gaat u beide in de *File Manager* van de Class in Möbius uploaden om ernaar te kunnen verwijzen als de procedures gebruikt worden in de vraag.

Als u veranderingen of aanvullingen in de procedures maakt, dan moet u de nieuwe .lib en .ind bestanden naar de *File Manager* in de Class uploaden.

1.2.2 Bestanden uploaden in de file manager

Hebt u de procedures gemaakt in Maple en zijn de .lib en .ind bestanden gegenereerd, die weggeschreven zijn naar dezelfde directory waarin ook het Maple Worksheet is opgeslagen, dan gaat u deze bestanden nu naar de *File Manager* uploaden.

Het is handig als u ook het worksheet zelf in dezelfde map upload, zodat u later aanpassingen kunt doen uitgaande van het Maple Worksheet en de nieuwe .lib en .ind bestanden weer kunt genereren en opnieuw uploaden.



Figure 1.66: File manager met repository bestanden

1.2.3 Gebruik van de bestanden in het algorithm

Dan komt nu het maken van de vraag in Möbius waar gebruikgemaakt kan worden van de procedures die opgeslagen zijn met vermelding waar ze opgeslagen zijn.

Het gemakkelijkste is het om in het *Algorithm* van de vraag te werken:

```
$i:=range(1,41);
$rij:=maple("MatrixInlezen[RijM]($i), libname=...../
MatrixInlezen.lib");
$a:=maple("$rij[1]");
$b:=maple("$rij[2]");
$som=$a+$b;
```

TIP: Als u niet weet wat het adres is van de lib bestanden, dan kunt u gebruikmaken van de *Designer* in het *Algorithm*. Via de knop *Maple Repository* kunt u in de *File Manager* het betreffende .lib-bestand aanklikken (zie figuur hieronder).

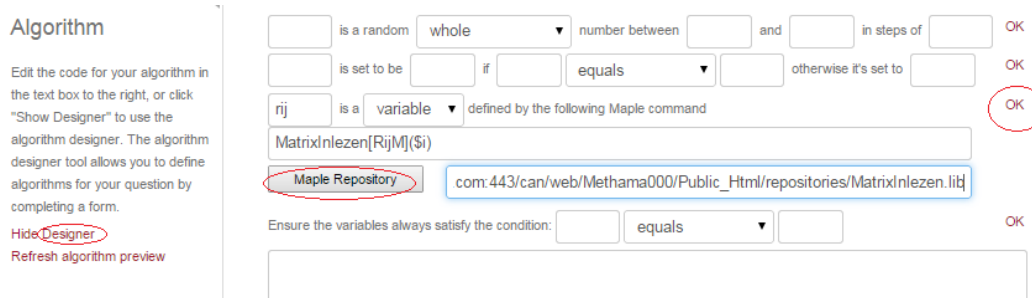


Figure 1.67: Het .lib-bestand ophalen met de Designer van het Algorithm

Vergeet vooral niet op *OK* te klikken in de *Designer*, tenzij u deze alleen aandoet voor het achterhalen van het adres van het lib-bestand.

1.3 Intervallen en ongelijkheden met Maple-repository*

Er is een goede set Maple-routines gemaakt die u kunt gebruiken voor intervallen en ongelijkheden.

Om te beginnen hebt u de *Maple Repository* nodig die u in de *File Manager* zet.

Het zijn twee bestanden. Zet ze in een mapje:

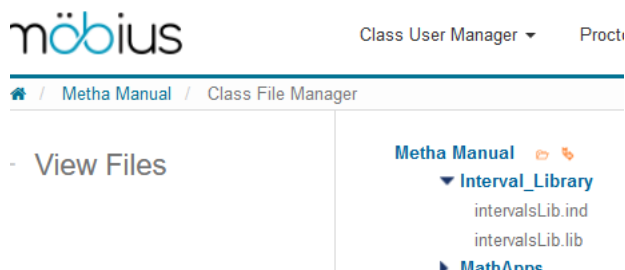


Figure 1.68: Interval files in the File Manager

U krijgt deze bestanden bij de bijbehorende cursus Möbius voor gevorderden bij CAN-diensten.

Start with a simple Question asking to type the interval $(3,5]$.

Use the Maple Graded question type.

Enter the following interval: (3,5]

Maple

Edit Response Area

Question

Feedback

Maple-graded:

Weighting: 1

Answer: `printInterval("(3,5]");`
(referenced when grading as \$ANSWER)

Grading Code: `gradeInterval("$RESPONSE", "(3,5]");`

Expression Type: Maple Syntax - e.g. `diff(2*f(x),x)`

Text/Symbolic entry: Text entry only

Optional:

Maple Repository: `https://can.mobius.cloud/web/Methama001/Public`
Maple Repository

Plotting Code:

Custom Previewing Code: `printInterval("$RESPONSE");`

Cancel

Update

Figure 1.69: Maple grade interval

For all the things in the dialog you need commands from the Repository prepared in Maple:

- The correct answer you put in the *Answer* section: `printInterval("(3,5]");`
- The *Grading Code* is with the command: `gradeInterval("$RESPONSE", "(3,5]");`
- *Expression Type*: Choose *Maple Syntax* with *Text entry only*.
- *Maple Repository*: Choose the correct file in the *File Manager* by pressing on the button *Maple Repository*.
You will find the `.lib` file in the *File Manager*.
- *Custom Previewing Code*: `printInterval("$RESPONSE");`

HINT: Before you Update this answer field first copy the url of the `.lib` file and put it in the *Algorithm* as a variable.

In the *Algorithm* you can program some more with randomization.

Start with the `$lib` so you can use it to prepare the plots with the command `drawInterval`.

```
$lib="...../Interval_Library/intervalsLib.lib";
$a=-1;
$b=range(6,7);
$index1=rnt(3);
```

```

$index2=rint(4);
$index3=rint(4);
$index4=rint(6);
$ans1=switch($index1,"(-infinity,$a)","($a,infinity)","(-infinity,
$a]","[$a,infinity)");
$anso1=switch($index1,"x < $a","x > $a","x<=$a","x>=$a");
$pl1=plotmaple("drawInterval(`$ans1`, span=-3..12, style=point),
  plotoptions='width=300,height=75', libname='$lib'");
$test=plotmaple("drawInterval(`$ans1`,
  plotoptions='width=300,height=75', libname='$lib'");

```

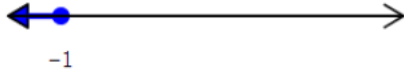
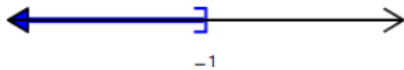
Variable	Value	Range
lib	/web/Methama001/Public_Html/Interval_Library/intervalsLib.lib	
a	-1	
b	6	6 - 7
index1	2	0 - 2
index2	0	0 - 3
index3	3	0 - 3
index4	1	0 - 5
ans1	$(-\infty, -1]$	
anso1	$x \leq -1$	
pl1		
test		

Figure 1.70: Algorithm

HINT: Mind that the interval and also the inequality are strings. So if you use the string `$ans1`, then in the command `drawInterval` use exactly this string: between back quotes ``$ans1``.

HINT: Put the name of the *Maple Repository* between normal quotes: `libname='$lib'`.

HINT: There are two ways of plotting this interval in a plot, with or without `style=point`, what you prefer. The plot you can use in the *Question Feedback*.

For the grading of an inequality you do something similar:

Enter the following inequality: \$anso1

Maple

Edit Response Area

Question Feedback

Maple-graded:

Weighting: 1

Answer: printInequality("\$anso1")
(referenced when grading as \$ANSWER)

Grading Code: gradeInequality("\$RESPONSE", "\$anso1");

Expression Type: Maple Syntax - e.g. diff(2*f(x),x) ▾

Text/Symbolic entry: Text entry only ▾

Optional:

Maple Repository: https://can.mobius.cloud/web/Methama001/Public
Maple Repository

Plotting Code:

Custom Previewing Code: printInequality("\$RESPONSE");

Figure 1.71: Maple graded inequality

For all the things in the dialog you need commands from the Repository prepared in Maple:

- The correct answer you put in the *Answer* section: `printInequality("$anso1");`
- The *Grading Code* is with the command:
`gradeInequality("$RESPONSE", "$anso1");`
- *Expression Type*: Choose *Maple Syntax* with *Text entry only*.
- *Maple Repository*: Choose the correct file in the *File Manager* by pressing on the button *Maple Repository*.
You will find the `.lib` file in the *File Manager*.
- *Custom Previewing Code*: `printInequality("$RESPONSE");`

HINT: For the student he has to know how to deal with intervals and inequalities.

1.4 Tweektalige vraag

Maak de tekst in de vraag tweektalig op de volgende manier:

Tik steeds twee talen vlak na elkaar en ga in de broncode de html-tags aanpassen op de volgende manier:

```
<p class="nl">Tik de formule na.</p>
<p class="en" style="display: none;">Enter the following formula.</p>
```

De extra optie `style="display: none;"` zet u steeds bij de engelse versie als u de nederlandse taal default wilt hebben.

Dus overall de tekst in twee talen steeds tussen de tags. Als u klaar bent met alles, dan de vraag finishen en alles controleren.

Open nu de broncode van de vraag met de knop *Edit Source*.

Om de twee knopjes voor de talen aan te maken moet u het volgende script inladen in de broncode.

Er worden dan twee knopjes aangemaakt om te kiezen tussen de twee talen. Daarna finishen en verder niet meer aankomen.

```
<p class="nl"><font size="4"><strong>Nederlands</strong></font></p>
<p class="en" style="display: none;"><font size="4"><strong>English</strong></font></p>
<p align="right">&nbsp;<button onclick="var
  enTXT=document.getElementsByClassName('en');var
  nlTXT=document.getElementsByClassName('nl');
  for(var i=0;i<enTXT.length;i++)
  {enTXT[i].style.display='block';nlTXT[i].style.display='none';}return false;">EN</
  button><button onclick="var enTXT=document.getElementsByClassName('en');var
    nlTXT=document.getElementsByClassName('nl');
    for(var i=0;i<enTXT.length;i++)
    {enTXT[i].style.display='none';nlTXT[i].style.display='block';}return false;">NL</
  button></p>
```

TIP: Nadat u de bovenstaande tekst hebt geplakt in de broncode, haal de regelovergangen dan weg en let op spaties in het script. Anders werkt het niet goed.

Als u toch nog iets in de vraag moet aanpassen, doe dat dan in de broncode en anders de vraag gewoon aanpassen en later in de broncode opnieuw het script toevoegen en het oude verwijderen. Sommige quotes gaan namelijk niet goed mee en dan wordt het script verminkt.

TIP: In een *Multiple Choice-Response Area* kunt u in de broncode van ieder alternatief de twee talen invoegen op de volgende manier:

```
<p class="nl">ja</p>
<p class="en" style="display: none;">yes</p>
```

Bij het vraagtype *List* gaat dat niet.

TIP: Voor hints werkt de tweektaligheid niet.

1.5 Kansrekenen en statistiek

Hieronder staan een paar commando's genoemd die in aanmerking komen voor gebruik bij de kansberekeningen met Möbius.

Statistics[CDF] = cumulative density function
 Statistics[Quantile] = inverse van de cumulatieve density function
 Statistics[ProbabilityFunction] = kansfunctie

Deze commando's kunt u gebruiken bij allerlei verdelingen zoals Normale verdeling, Poissonverdeling of andere verdelingen zoals in de volgende paragrafen wordt besproken.

Voor het manipuleren van data in het kader van beschrijvende statistiek zijn het de commando's:

Statistics[Mean]
 Statistics[Mode]
 Statistics[Range]
 Statistics[Median]

 Statistics[Variance]
 Statistics[StandardDeviation]

TIP: Kijk ook eens naar de Maple commando's in het pakket Student[Statistics].

1.5.1 Normale verdeling

$N(\mu, \sigma)$ = Normale verdeling (variabele x)

$N(0, 1)$ = Standaardnormale verdeling (variabele z)

De kans $P(x \leq 1)$ dat bij een normaal verdeelde kansvariabele met $\mu = 3$ en $\sigma = 2$ de x -waarde kleiner is of gelijk aan 1.

We gebruiken hier de functie CDF de cumulative density function.

In figuur *Figure 1.72 (page 81)* is te zien hoe deze commando's in de rubriek *Algorithm* gebruikt kunnen worden.

```
> Statistics[CDF](Normal(3,2),1,numeric);
```

0.158655253931457 (1.5)

TIP: Als u de optie *numeric* weglaat, dan geeft Maple het exacte antwoord geformuleerd met de functie erf.

Echter u kunt in plaats van het getal 1 ook 1.0 invoeren en dan is de optie *numeric* niet meer nodig.

De kans is 68% dat de variabele tussen $-\sigma$ en σ zit.

```
> Statistics[CDF](Normal(0,1),1,numeric)-Statistics[CDF]
(Normal(0,1),-1,numeric);
```

0.682689492137086 (1.6)

De kans is 95.5% dat de variabele tussen -2σ en $+2\sigma$ zit.

```
> Statistics[CDF](Normal(0,1),2.0)-Statistics[CDF](Normal(0,1),-2.0);
```

0.954499736103642 (1.7)

De grenswaarde g waarvoor geldt dat $P(z \leq g) = 0.6$ bij standaard normale verdeling.

We gebruiken hierbij de functie Statistics[Quantile] (de inverse cumulative density function) en kijken in het programma Maple wat het effect is.

Als duidelijk is wat we willen, kan deze functionaliteit gebruikt worden in de rubriek *Algorithm* om variabelen aan te maken.

```
> g:=Statistics[Quantile](Normal(0,1),0.6);
```

$$g := 0.253347103135800 \quad (1.8)$$

```
> g1:=Statistics[Quantile](Normal(3,2),0.6);
```

$$g1 := 3.50669420627160 \quad (1.9)$$

Inderdaad is de grenswaarde $g1$ t.o.v. de vorige grenswaarde g verschoven met 3 eenheden. Bovendien is de sigma tweemaal zo groot geworden, zodat het antwoord van de grenswaarde $g1$ in de lijn der verwachting ligt.

TIP: Er hoeft nu niet de optie *numeric* bijgegeven te worden want Maple rekt automatisch numeriek als het argument numeriek is.

TIP: Behalve dat u Maple het werk kunt laten doen voor deze acties, kunt u ook voor veel dingen direct het systeem gebruiken met de functie $\text{erf}((x-\mu)/\sigma)$ en de inverse daarvan inverf . Echter deze laatste laat zich alleen gebruiken voor de standaard normale verdeling. Met $\text{inverf}(0.6)$ krijgt u dus de z -waarde 0.253.

Bovenstaande berekeningen kunnen dan als volgt gedaan worden in het *Algorithm*:

```
$test1:=erf((1-3)/2);
$Test1:=maple("Statistics[CDF](Normal(3,2),1,numeric)");
$test2:=erf(1)-erf(-1);
$Test2:=maple("Statistics[CDF](Normal(0,1),1,numeric)-Statistics[CDF](Normal(0,1),-1,numeric)");
$test3:=erf(2)-erf(-2);
$Test3:=maple("Statistics[CDF](Normal(0,1),2.0)-Statistics[CDF](Normal(0,1),-2.0)");
$test4:=inverf(0.6);
$Test4:=maple("Statistics[Quantile](Normal(0,1),0.6)");
$Test5:=maple("Statistics[Quantile](Normal(3,2),0.6)");
```

Variable	Value
test1	0.158655
Test1	.158655253931457
test2	0.682689
Test2	.682689492137086
test3	0.9545
Test3	.954499736103642
test4	0.253347
Test4	.253347103135800
Test5	3.50669420627160

Figure 1.72: Normale verdeling

Voor een **normaalverdeelde sample** maak je in het algoritme het volgende programma met commando's uit het Student[Statistics]-pakket :

```
$mu:=range(8,15);
$sigma:=range(2,4);
$Sample:=maple("with(Student[Statistics]):randomize():P:=NormalRandomVariable($mu,
$sigma):evalf(convert(Sample(P,10),list),3)");
```

```

$lijst=maple("with(Student[Statistics]):
[Mean($Sample),Median($Sample),Variance($Sample),StandardDeviation($Sample),Quartile($Sample,1),Quartile($Sample,3)
$mean=switch(0,$lijst);
$median=switch(1,$lijst);
$variance=decimal(3,switch(2,$lijst));
$standarddev=decimal(2,switch(3,$lijst));
$q1=switch(4,$lijst);
$q3=switch(5,$lijst);

```

Stel eerst een gemiddelde en standaarddeviatie vast.

Genereer dan een saple van 10 getallen (of meer of minder) die je afrond op 3 significante cijfers.

Maak dan een lijst met de verschillende grootheden die je wilt overhoren.

Omdat de lijst bestaat uit afgeronde getallen is mu en sigma dus niet meer de oorspronkelijke waarde.

De student berekent met de rekenmachine en rondt af.

1.5.2 Binomiale verdeling

Binomiale verdeling $\text{Binomial}(n, \pi)$. Met n het aantal pogingen en π de kans op succes.

De kans op precies $k = 4$ successen $P(k = 4)$ bij een binomiale verdeling met $n = 15$ en $\pi = 0.7$.

Gebruik dan de kansfunctie: ProbabilityFunction .

```

> Statistics[ProbabilityFunction](Binomial(15,0.7),4);
0.0005805753777

```

(1.10)

```

> Statistics[ProbabilityFunction](Binomial(15,0.7),1);
5.02211745 10-7

```

(1.11)

De kans $P(k \leq 4)$ op 4 successen of minder van de 15 pogingen bij een binomiale verdeling met $n = 15$ en $\pi = 0.7$.

Gebruik dan de functie CDF cumulative density function.

```

> Statistics[CDF](Binomial(15,0.7),4);
0.0006722340698

```

(1.12)

Deze zelfde functie CDF kan ook gebruikt worden voor: $P(k=4)$:

```

> Statistics[CDF](Binomial(15,0.7),4)-Statistics[CDF]
(Binomial(15,0.7),3);
0.0005805753776

```

(1.13)

Kans op 11 pech (kans op pech = 0.3) of meer van de 15 pogingen is gelijk aan 1 - de kans op 10 pech of minder.

```

> 1-Statistics[CDF](Binomial(15,0.3),10);
0.0006722341

```

(1.14)

Kans op 4 succes of meer = $P(k \geq 4) = 1 - P(k \leq 3)$.

```

> 1-Statistics[CDF](Binomial(15,0.7),3);
0.9999083413

```

(1.15)

U kunt ook doen de kans op pech = 0.3 en dan is de kans op pech dus $P(\text{pech} \leq 11)$ als u wilt berekenen de kans op 4 succes of meer.

```
> Statistics[CDF](Binomial(15,0.3),11);
```

$$0.9999083413 \quad (1.16)$$

Als de kans 0.5 is, wat was dan de waarde van k ? bij een binomiale verdeling met $n = 15$ en $\pi = 0.7$.

Quantile is de inverse cumulative probability function. (Let op in dit geval discrete waarden met discontinuïteit.)

```
> Statistics[Quantile](Binomial(15,0.7),0.5);
```

Dus $P(k \leq 11)$ is 0.5 bij een binomiaalverdeling met 15 pogingen en 0.7 kans op succes.

TIP: Möbius bevat ook de binomiaalfunctie $\text{binomial}(n, k)$ maar hiermee kan slechts het binomium berekend worden bijvoorbeeld $\binom{n}{k}$ om bijvoorbeeld te berekenen op hoeveel manieren k objecten gekozen kunnen worden uit een verzameling van n objecten.

1.5.3 Poisson verdeling

De Poissonverdeling heeft maar één parameter: $\text{Poisson}(\mu)$.

De kans $P(k=2)$ bij een poissonverdeling met $\mu = 3$ berekenen we als volgt:

```
> Statistics[ProbabilityFunction](Poisson(3),2);
```

$$\frac{9}{2} e^{-3} \quad (1.17)$$

Dit is precies de exacte waarde zoals de kansformule van Poisson voorschrijft.

Als u de numerieke waarde wilt, geeft u de optie numeric mee of u schrijft voor het getal 2 een decimaal getal 2.0.

```
> Statistics[ProbabilityFunction](Poisson(3),2.0);
```

$$0.2240418077 \quad (1.18)$$

De kans ($P(k \leq 2)$) bij een Poissonverdeling met $\mu = 3$ berekent u met de cumulatieve kansdichtheidsfunctie:

```
> Statistics[CDF](Poisson(3),2.0);
```

$$0.423190081126844 \quad (1.19)$$

De inverse van de cumulatieve kansdichtheid is weer Quantile:

```
> Statistics[Quantile](Poisson(3),0.42);
```

$$2. \quad (1.20)$$

Als de kans gelijk is aan 0.42 dan is dat de kans $P(k \leq 2)$ bij $\mu = 3$ van Poisson (Let wel op dat dit een discrete kansverdeling is dus discontinu).

1.5.4 Students_t-verdeling

Students_t wordt gebruikt bij steekproeven als μ en σ van een verdeling onbekend zijn om betrouwbaarheidsintervallen aan te geven.

Als x (gemiddelde waarde van de steekproef) een schatter is van μ , en s (standaarddeviatie van een steekproef) is een schatter van σ van de populatie, dan is het betrouwbaarheidsinterval:

Zie voor deze schatters in paragraaf Data (page 85)

$$x - \frac{ts}{\sqrt{n}} < \mu < x + \frac{ts}{\sqrt{n}}$$

De **students_t[v]** heeft dus één parameter en dat is v het aantal vrijheidsgraden is gelijk aan $n - 1$ als n de steekproefgrootte is.

> Typesetting: -RuleAssistant();

Als de steekproef groter is dan 30, kan de normale verdeling gebruikt worden.

De kans dat bij een steekproef van 5 stuks (aantal vrijheidsgraden = 4), de t -waarde kleiner is of gelijk aan 2, is gelijk aan $P(t \leq 2)$.

> kans:=Statistics[CDF](StudentT(4),2.0);

kans := 0.941941738241594 (1.21)

Het 95% betrouwbaarheidsinterval wordt bepaald door de volgende t -waarde met een steekproefgrootte van $n = 5$.

> t:=Statistics[Quantile](StudentT(4),0.975);

t := 2.77634649058963 (1.22)

Voor een steekproef van 5 stuks is het aantal vrijheidsgraden $v = 4$ en geldt dat het betrouwbaarheidsinterval tussen $+2.776$ en -2.776 maal de "sigma" verwijderd ligt van het gemeten gemiddelde van de steekproef, bij een rechteroverschrijdingskans van 0.975 wat correspondeert met een 95% betrouwbaarheidsinterval.

TIP: Möbius bevat ook de students_t-kansdichtheidsfunctie **students(v, x)** en de inverse ervan **invstudentst(v, P)**.

```
$test1=studentst(4,2);
$Test1=maple("Statistics[CDF](StudentT(4),2.0)");
$test2=invstudentst(4,0.975);
$Test2=maple("Statistics[Quantile](StudentT(4),0.975)");
```

Variable	Value
test1	0.941942
Test1	.941941738241594
test2	2.776445
Test2	2.77634649058963

Figure 1.73: De Students_t-verdeling

1.5.5 χ^2 -verdeling

De χ^2 -verdeling

Bij een berekende waarde van χ^2 van 10.5 bij een situatie van 4 vrijheidsgraden is de kans $\alpha = (100 - 96.7)\% = 3.3\%$ dat de verschillen ontstaan door toevalligheden.

```
> Kans:=Statistics[CDF](ChiSquare(4),10.5);
```

$$Kans := 0.967203010005116 \quad (1.23)$$

Kritieke grens voor χ^2 bij een overschrijdingskans van 0.05.

```
> kritieke_grens=Statistics[Quantile](ChiSquare(4),0.95);
```

$$kritieke_grens = 9.48772903678116 \quad (1.24)$$

Als $\chi^2(4)$ dus groter is dan 9.4877, dan is de kans dat de afwijkingen op toeval berusten 5%.

```
> kritieke_grens=Statistics[Quantile](ChiSquare(4),0.99);
```

$$kritieke_grens = 13.2767041359876 \quad (1.25)$$

Als $\chi^2[4]$ groter is dan 13.2767, dan is de kans dat de afwijkingen op toeval berusten nog maar 1%.

1.5.6 Data

Hieronder volgen enkele tools om in voorkomende gevallen met data om te gaan.

```
> restart; data:=[6.7,8.5,11.8,9.4,7.6,9.2,10,8.8,10,12,11,6,8];
```

$$data := [6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8] \quad (1.26)$$

De operanden van de lijst:

```
> op(data);
```

$$6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8 \quad (1.27)$$

De minimumwaarde:

```
> min(data);
```

$$6 \quad (1.28)$$

De maximumwaarde:

```
> max(data);
```

$$12 \quad (1.29)$$

Gemiddelde waarde:

```
> Statistics[Mean](data);
```

$$9.15384615384615 \quad (1.30)$$

De modus (welke meting het vaakst voorkomt)

```
> Statistics[Mode](data);  
  
10. (1.31)
```

De Range (verschil tussen hoogste en laagste waarde)

```
> Statistics[Range](data);  
  
6. (1.32)
```

De mediaan (middelste waarneming nadat alles op volgorde gezet is)

```
> Statistics[Median](data);  
  
9.200000000000000 (1.33)
```

Eerste, tweede en derde kwartiel:

```
> Statistics[Quartile](data,1);  
  
7.866666666666667 (1.34)
```

```
> Statistics[Quartile](data,2);  
  
9.200000000000000 (1.35)
```

```
> Statistics[Quartile](data,3);  
  
10.333333333333333 (1.36)
```

```
> Statistics[Decile](data,1);  
  
6.466666666666667 (1.37)
```

```
> Statistics[Decile](data,5);  
  
9.200000000000000 (1.38)
```

```
> Statistics[Decile](data,6);  
  
9.600000000000000 (1.39)
```

Variantie van de steekproef:

```
3.356025641 (1.40)
```

```
> Statistics[Variance](data);  
  
3.35602564102564 (1.41)
```

Standaarddeviatie van de steekproef.

```
> Statistics[StandardDeviation](data);  
  
1.83194586192541 (1.42)
```

```
> sort(data);  
  
[6, 6.7, 7.6, 8, 8.5, 8.8, 9.2, 9.4, 10, 10, 11, 11.8, 12] (1.43)
```

```
> Statistics[Count](data);
```

```
13
```

 (1.44)

```
> data;
```

```
[6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8]
```

 (1.45)

```
> lijst:=[[3,4,5],[6,7,8]];
```

```
lijst := [[3, 4, 5], [6, 7, 8]]
```

 (1.46)

```
> lijst[1];
```

```
[3, 4, 5]
```

 (1.47)

1.5.7 Visualisaties

Een voorbeeld van een grafiek gebaseerd op data:

Maak eerst een lijst of een matrix met data. Vervolgens kunt u in het *Algorithm* de grafiek voorbereiden als variabele.

```
$data:=maple("[6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8]");
```

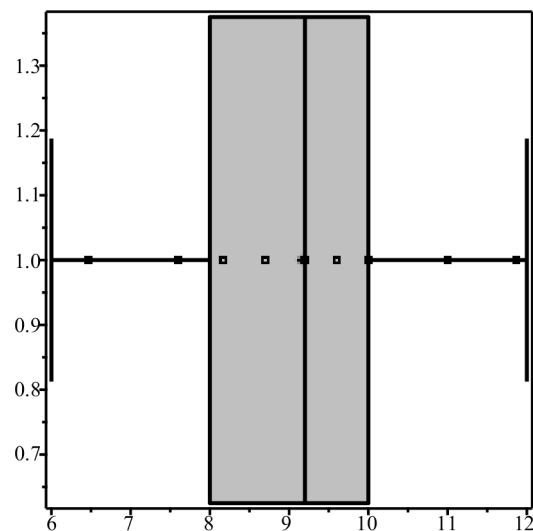
```
$boxplot=plotmaple("Statistics[BoxPlot]($data),plotoptions='height=300, width=300'");
```

Echter u kunt met extra opties bij het plotcommando de figuur nog verder aanpassen.

Het is ook mogelijk van grotere datasystemen een gecombineerd boxplot te maken.

```
> data:=[6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8];
Statistics[BoxPlot]
(data, orientation=horizontal, deciles=true, color=gray, axes=boxed, outliers=true);
```

```
data := [6.7, 8.5, 11.8, 9.4, 7.6, 9.2, 10, 8.8, 10, 12, 11, 6, 8]
```

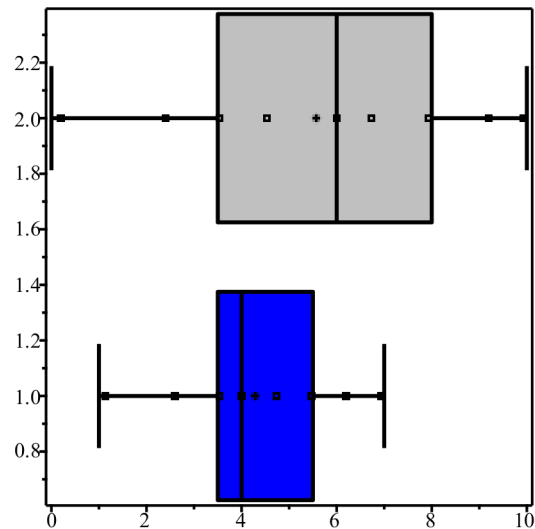


```
> M:=Matrix([[ 1 , 4 ],
[ 4 , 6 ],
[ 5 , 9 ],
[ 7 , 10 ],
[ 4 , 3 ],
[ 3 , 0 ]],
```

```
[ 6 , 7 ]]);
```

$$M := \begin{bmatrix} 1 & 4 \\ 4 & 6 \\ 5 & 9 \\ 7 & 10 \\ 4 & 3 \\ 3 & 0 \\ 6 & 7 \end{bmatrix} \quad (1.49)$$

```
> Statistics[BoxPlot]
(M,orientation=horizontal,deciles=true,color=[blue,gray],axes=boxed,outliers=true);
```



1.6 Statistiek

Kies voor vraagstukken met kansen altijd voor een *Numeric* antwoordveld waarbij numerieke instellingen mogelijk zijn.

Bijvoorbeeld met margin of error = 0.0001 of anders.

Studenten rekenen de kans uit met rekenmachine of tabellenboek meestal in vier decimalen.

In de rubriek *Algorithm* gaat u het antwoord voorbereiden, eventueel met behulp van de functionaliteit van Maple.

Bij de berekening met Maple komen altijd veel decimalen in het antwoord.

Als u wilt dat een beperkt aantal decimalen wordt toegestaan, kan dat in tweede instantie geregeld worden met `decimal(4, ...)`.

Als u Maple het werk laat doen voor statistische berekeningen, dan maakt u gebruik van het pakket *Statistics* met daarin een aantal commando's die u kunt gebruiken. Het is een nieuw pakket waarin allerlei zaken die te maken hebben met dit onderwerp goed georganiseerd zijn.

1.6.1 Beschrijvende statistiek

Met data analyse en visualisaties zoals histogram en boxplot

1.6.1.1 Datasets genereren

Permutatie van een lijst met getallen

Begin met een lijst met een aantal willekeurige getallen.

Met het commando `randperm` uit het `combinat`-pakket maakt u daarvan een permutatie.

Door bij ieder element (met `~`) van de lijst een getal `$d` erbij op te tellen, krijgt u een nieuwe lijst.

Neem (met het commando `op`) de operanden van de lijst om de rij getallen te krijgen.

Met het commando `sort` zet u de elementen van de lijst op opklimmende volgorde.

Zie daarvoor het volgende *Algorithm*:

```
$data1=maple("[12, 13, 23, 45, 14, 13, 23, 48, 13, 45]");
$d=range(5, 20, 1);
$data2=maple("($data1)+~$d");
$data=maple("randomize():StringTools[Randomize]
():combinat[randperm]($data2)");
$Ddata=maple("op($data)");
$gesort=maple("op(sort($data))");
```

Variable	Value	Range
data1	[12, 13, 23, 45, 14, 13, 23, 48, 13, 45]	
d	12	5 - 20
data2	[24, 25, 35, 57, 26, 25, 35, 60, 25, 57]	
data	[35, 57, 26, 60, 24, 25, 25, 57, 35, 25]	
Ddata	35, 57, 26, 60, 24, 25, 25, 57, 35, 25	
gesort	24, 25, 25, 25, 26, 35, 35, 57, 57, 60	

Figure 1.74: Permutatie van een lijst getallen

TIP: Met de commando's `randomize():StringTools[Randomize]()`: vooraf zorgt u ervoor dat u iedere keer een nieuwe permutatie krijgt. (Zie ook in de handleiding *Randomiseren*.)

Rij willekeurige getallen

In het volgende wordt een rij van een aantal getallen random gegenereerd met het commando

Generate uit het `RandomTools`-pakket. Het is een willekeurig aantal `$n` (tussen 7 en 12)

getallen die liggen in een range van 10 tot 80.

In het *Algorithme* ziet u hoe de getallen van deze lijst stuk voor stuk met 100 vermenigvuldigd

worden en vervolgens afgerond om daarna met 10^{-2} te vermenigvuldigen zodat u getallen krijgt met twee decimalen.

Met het opvragen van alle operanden van de verkregen lijst, krijgt u een rij getallen (en dus geen lijst meer).

Eventueel ten behoeve van de feedback is de lijst met getallen op volgorde te zetten met het commando `sort`.

Het commando **Median** uit het `Statistics`-pakket vereist dat u met een lijst werkt en niet met een rij getallen.

U ziet het in het volgende *Algoritme*.

```
$n=range(7, 12);
$datafl=maple("randomize():
RandomTools[Generate](list(float(range=10..80), $n))");
$data=maple("map(x->Float(round(100*x), -2), $datafl)");
$datageg=maple("op($data)");
```

```
$datagesorteerd=maple("sort($data)");
$mediaan=maple("Statistics[Median]($data)");
```

Variable	Value	Range
n	8	7 - 12
datafl	[21.97240803, 45.74434317, 47.64912527, 14.41185866, 78.96055853, 62.07103014, 72.33983329, 34.80918783]	
data	[21.97, 45.74, 47.65, 14.41, 78.96, 62.07, 72.34, 34.81]	
datageg	21.97, 45.74, 47.65, 14.41, 78.96, 62.07, 72.34, 34.81	
datagesorteerd	[14.41, 21.97, 34.81, 45.74, 47.65, 62.07, 72.34, 78.96]	
mediaan	46.6950000000000	

Figure 1.75: Data genereren willekeurig

TIP: De extra opdracht `randomize()` is ervoor om elke keer weer andere getallen te krijgen. (Zie ook in de handleiding *Randomiseren*.)

Een steekproef uit een verdeling

In het volgende voorbeeld genereert u met het commando `Sample` uit het Statistics-pakket een steekproef die u meteen naar een lijst converteert. U krijgt dan een variabele `$test`: een lijst van `$N1` getallen getrokken uit een normale verdeling met gemiddelde `$g` en standaarddeviatie `$s`.

TIPS: Als u niet converteert naar een lijst, dan geeft het commando `Sample` uit het Statistics-pakket een rijvector en dat is niet altijd handig.

Van de gegenereerde lijst kunt u alle elementen (~) afronden met `round`. Maar u kunt ook alle elementen van de lijst (door middel van het commando `map`) afronden op een getal met twee decimalen met het commando `Float`.

Ten slotte kunt u (met het commando `op`) alle elementen van de lijst (operanden) achter elkaar krijgen in een rij (zonder haken eromheen).

Het *Algorithm* ziet er als volgt uit:

```
$g=switch(rint(5), "50", "80", "40", "60", "70");
$N1=range(8, 12);
$s=range(10, 15);
$test=maple("randomize():
convert(Statistics:-Sample(Normal($g, $s), $N1), list)");
$Alijst=maple("round~($test)");
$Blijst=maple("map(x->Float(round(100*x), -2), $test)");
$B=maple("op($Blijst)");
```

Het is ook mogelijk om data te genereren uit een andere verdeling.

Variable	Value	Range
g	80	
N1	12	8 - 12
s	14	10 - 15
test	[64.9860622080242, 75.3929098123411, 71.3607128832629, 83.0025344334341, 79.6440062074066, 104.203497978490, 57.1120054391853, 101.996410404560, 82.3850178997457, 94.0906977225253, 83.7619775082163, 59.4762737425771]	
Alijst	[65, 75, 71, 83, 80, 104, 57, 102, 82, 94, 84, 59]	
Blijst	[64.99, 75.39, 71.36, 83.00, 79.64, 104.20, 57.11, 102.00, 82.39, 94.09, 83.76, 59.48]	
B	64.99, 75.39, 71.36, 83.00, 79.64, 104.20, 57.11, 102.00, 82.39, 94.09, 83.76, 59.48	

Figure 1.76: Data genereren uit een verdeling

1.6.1.2 Staafdiagram

Staafdiagram

Met het commando `ColumnGraph` uit het Statistics-pakket maakt u een staafdiagram met verticale staven.

De tegenhanger is het horizontale staafdiagram dat u maakt met het commando `BarChart` uit het Statistics-pakket.

Standaard wordt een staafbreedte (`width`) van 0.75 aangehouden en de staven sluiten niet aan, maar hebben een standaard tussenruimte (`distance`) van 0.25.

Standaard beginnen ze bij 0, maar kan ook ingesteld worden met `offset`.

```
$a=range(40, 60, 5);
$b=range(30, 40, 5);
$c=range(20, 30, 5);
$d=range(10, 20, 5);
$e=range(5, 10, 5);
$data=maple("[$a, $b, $c, $d, $e]");
$plot=plotmaple("Statistics[ColumnGraph]($data,
tickmarks=[0.4=storing1, 1.4=storing2, 2.4=storing3, 3.4=storing4, 4.4=storing5], default
plotoptions='height=300, width=400'");
```

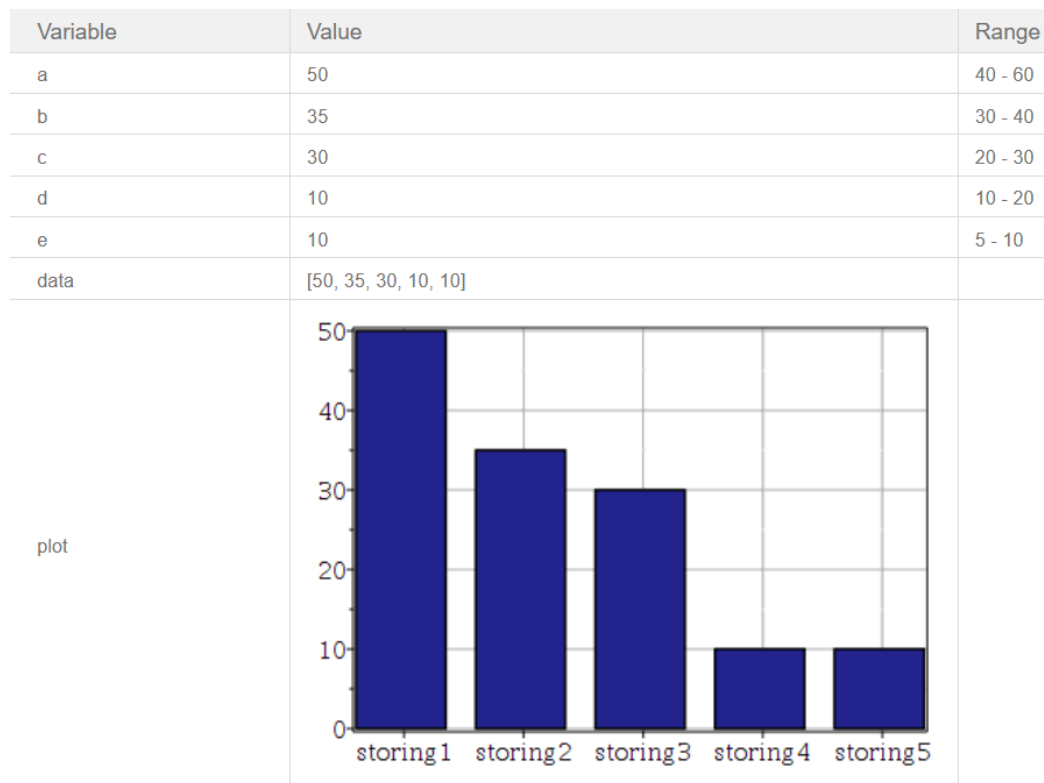


Figure 1.77: Staafdiagram

Paretodiagram

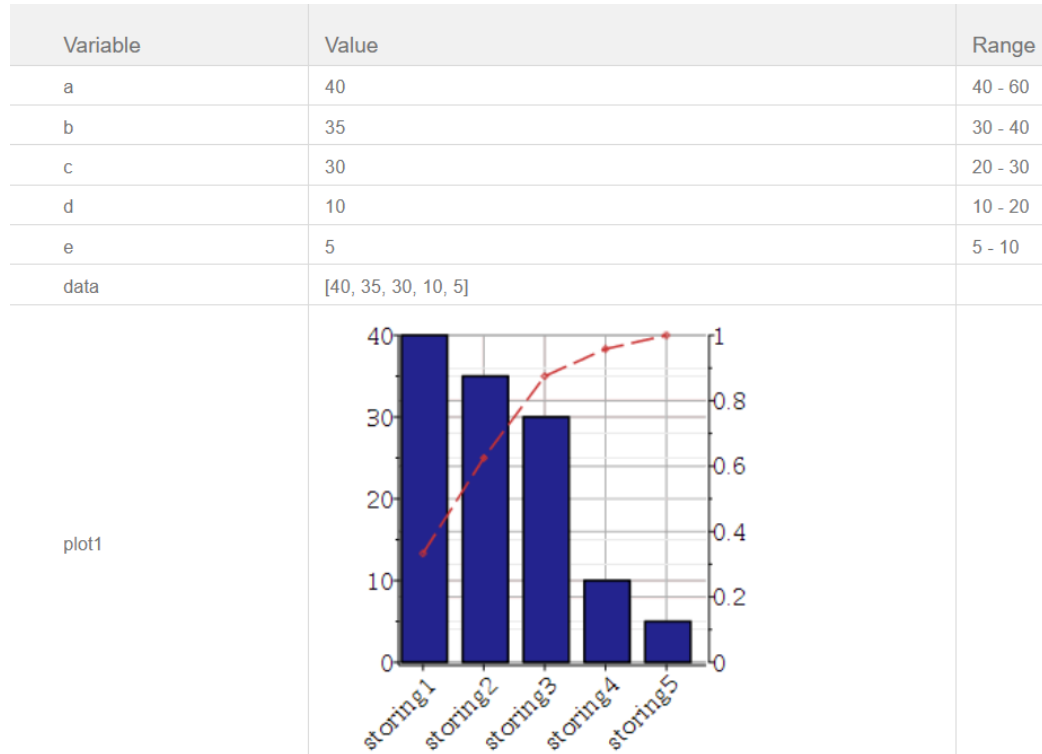
```
$a=range(40, 60, 5);
$b=range(30, 40, 5);
$c=range(20, 30, 5);
$d=range(10, 20, 5);
```

```

$e=range(5,10,5);
$data=maple("[a,$b,$c,$d,$e]");

$plot1=plotmaple("data:=DataSeries($data,labels=[`storing1`,`storing2`,`storing3`,`storing4`,`storing5`])
Statistics[ParetoChart](data,color=[navy,orange],gridlines=true,columngraphoptions
=[datasetlabels = none]),
plotoptions='height=300, width=300'");

```



1.6.1.3 Staafdiagram als histogram en cumulatief diagram

Uitgaande van een frequentieverdeling met totale frequentie 50 en een klassenverdeling met klassenbreedte 5, bereid u de lijst met **frequentiedichtheden** voor door elke frequentie van de lijst te delen door de klassenbreedte.

Een bijbehorend **histogram** maakt u dan met behulp van een staafdiagram waarbij de staven aaneengesloten zijn met de optie `distance=0`.

Het volgende *Algorithm* laat zien hoe u de frequentieverdeling met de frequentiedichtheden voorbereidt en er een histogram van maakt.

```

$ftotaal=50;
$w1=range(2,6);
$w2=range(6,8);
$w3=range(8,12);
$w5=range(8,12);
$w6=range(2,4);
$w7=range(1,2);
$wtot=$w1+$w2+$w3+$w5+$w6+$w7;
$w4=$ftotaal-$wtot;
$frequentie1="[$w1,$w2,$w3,$w4,$w5,$w6,$w7]";
$freqd1=maple("map(x->x/5,$frequentie1)");
$hist=plotmaple("Statistics[ColumnGraph]
($freqd1,scale=absolute,offset=15,width=5,distance=0,color=tan,labels=[`kg
`,`frequentiedichtheid`],labeldirections=[HORIZONTAL,VERTICAL]),plotoptions='height=200,
width=250'");
$upper="[15,20,25,30,35,40,45,50]";

```

```

$frequentie2=maple("ListTools[Flatten]([0,$frequentie1])");
$cumFreq1=plotmaple("plots[display]({Statistics[CumulativeSumChart]
($frequentie2, scale = relative,color = red, thickness
= 1, symbol = circle, symbolsize = 10,xcoords=
$upper)),gridlines=true,tickmarks=[default,20]),plotoptions='height=200,
width=250'");

```

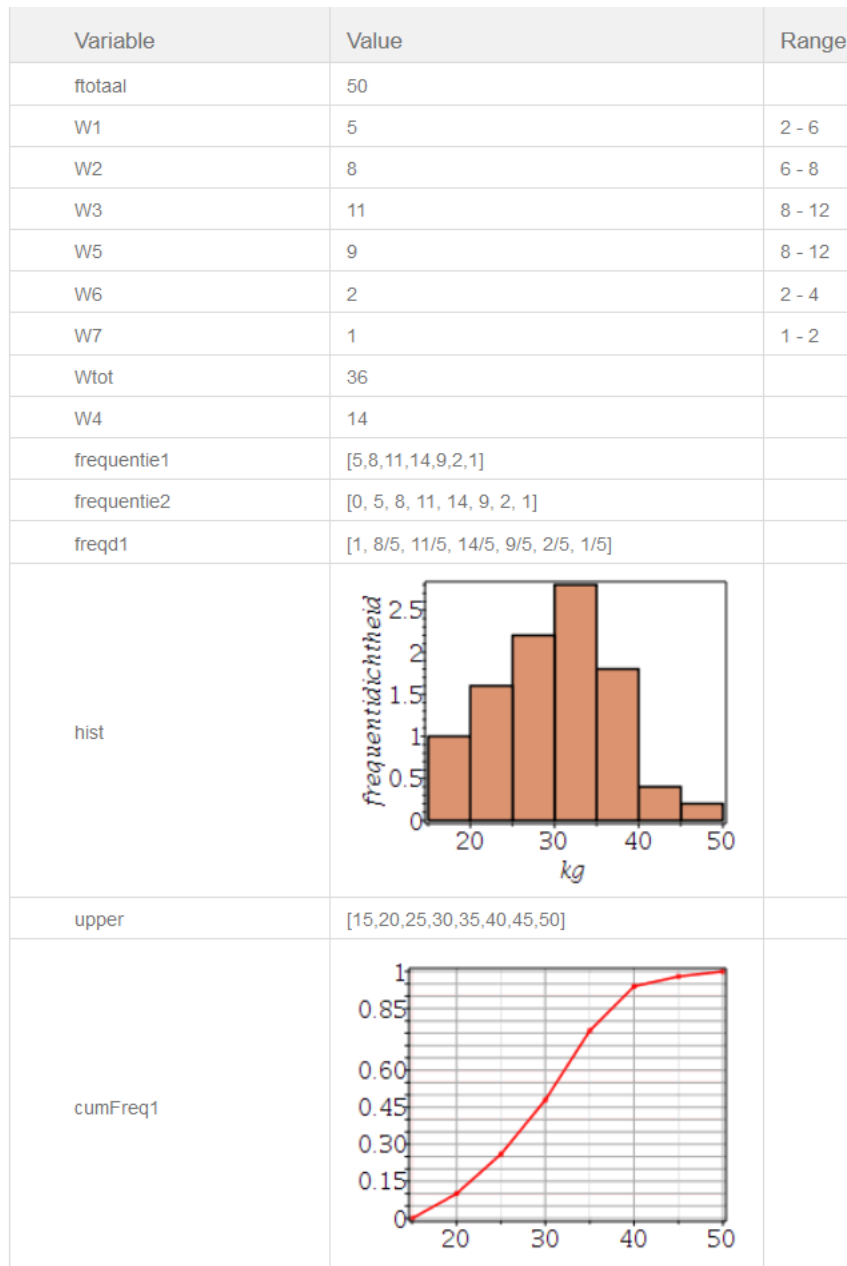


Figure 1.78: Staafdiagram als histogram

TIP: Ter voorbereiding van het cumulatieve frequentiediagram breidt u de lijst met frequenties uit naar links met behulp van het commando `Flatten` uit het `ListTools` pakket. Links van de staaf van de klasse 15-20 is de relatieve frequentie namelijk gelijk aan 0 zodat dit diagram begint bij 15 met frequentie 0. Met de optie `scale = relative` krijgt u de relatieve cumulatieve frequentieverdeling van de dataset.

1.6.1.4 Histogram van een dataset

In het volgende voorbeeld wordt een dataset gegenereerd van 100 randomgetallen (met twee decimalen) tussen 10 en 80. Eventueel kunt u de lijst met data op volgorde zetten met het commando `sort` (maar dat is niet noodzakelijk om het histogram te maken).

U kunt met een extra optie kiezen voor een aantal klassen bijvoorbeeld `bincount=7`, maar daarmee is niet gegarandeerd dat de klassen mooi verdeeld worden over de range (hier van 10 tot 80).

Met de optie `binbounds=[10, 20, 30, 40, 50, 60, 70, 80]` forceert u de indeling in klassen met mooie klassegrenzen.

De optie `frequencyscale=relative` is default en geeft de relatieve frequentiedichtheid langs de verticale as (zodat de totale oppervlakte van de staven gelijk is aan 1).

Met de optie `frequencyscale=absolute` bereikt u dat verticaal de absolute frequentie van iedere klasse wordt uitgezet.

```
$n=100;
$datafl=maple("randomize():
RandomTools[Generate](list(float(range=10..80),$n))");
$data=maple("map(x->Float(round(100*x), -2),$datafl)");
$datsort=maple("sort($data)");
$hist1=plotmaple("Statistics[Histogram]($data,
    bincount=7,color=tan,
    transparency=0.5,labels=[`kg`,`frequentidichtheid`],
    labeldirections=[HORIZONTAL,VERTICAL]),|
    plotoptions='height=200, width=250'");
$hist2=plotmaple("Statistics[Histogram]
($data,frequencyscale=absolute,
    binbounds=[10,20,30,40,50,60,70,80],color=blue,transparency=0.5,
    labels=[`kg
`,`frequentidichtheid`],labeldirections=[HORIZONTAL,VERTICAL]),
    plotoptions='height=200, width=250'");
```

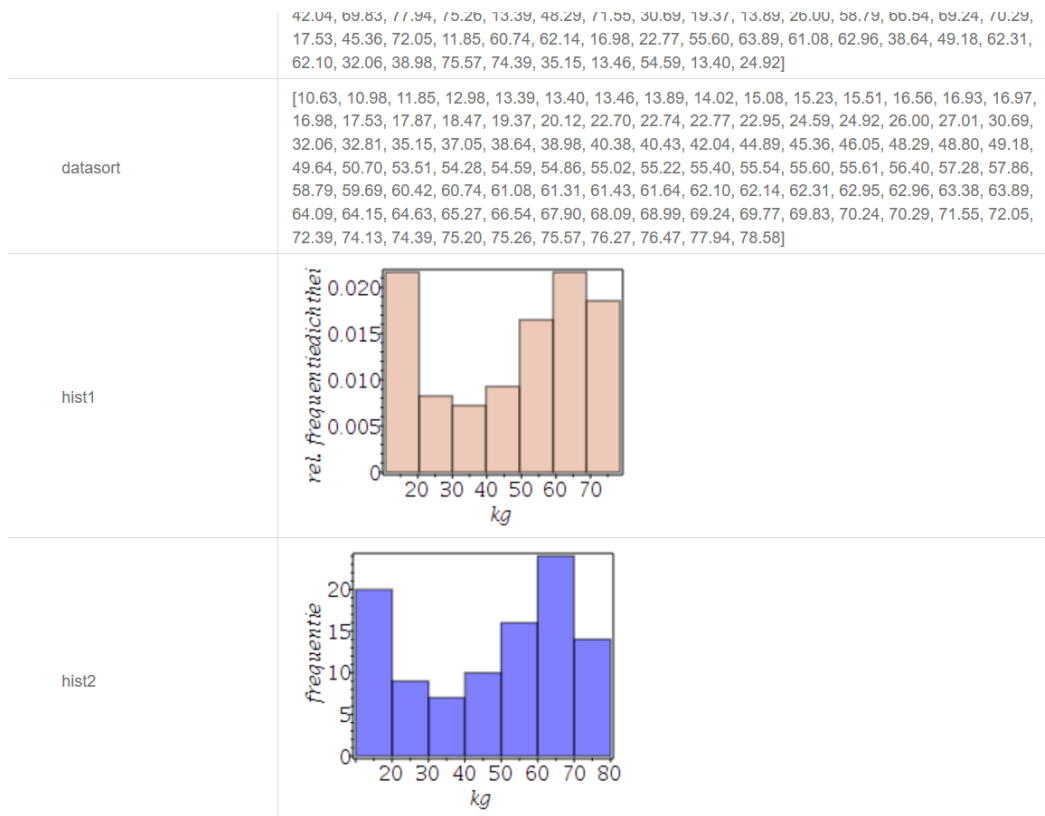


Figure 1.79: Histogram van dataset

1.6.1.5 Boxdiagram

In het volgende voorbeeld ziet u een figuur met één boxdiagram en een figuur met daarin 4 boxdiagrammen.

De uiteinden van de box wordt gevormd door het eerste en het derde kwartiel.

De streep middenin is de mediaan.

De stip staat voor het gemiddelde.

Het maximum en het minimum is aangegeven met de uiteinden van de twee stelen.

Eventueel kunnen uitschieters ook in beeld gebracht worden met de optie `outliers=true`.

Neem het volgende *Algorithm* over om de boxdiagrammen te maken

```
$gr=switch(rint(5),"50","80","40","60","70");
$N1=range(8,12);
$N2=range(8,12);
$N3=range(8,12);
$N4=range(8,12);
$s=range(10,15);
$A=maple("randomize():round~(convert(Statistics:-
Sample('Normal'($gr, $s), $N1), 'list'))");
$B=maple("randomize():round~(convert(Statistics:-
Sample('Normal'($gr, $s), $N2), 'list'))");
$C=maple("randomize():round~(convert(Statistics:-
Sample('Normal'($gr, $s), $N3), 'list'))");
```

```

$D=maple("randomize():round~(convert(Statistics:-
Sample('Normal'($gr, $s), $N4), 'list'))");
$p=plotmaple("Statistics[BoxPlot]
([$A],outliers=true,deciles=false,color=blue,
transparency=0.5,orientation=horizontal,tickmarks=[default,0]),
plotoptions='height=200, width=200'");
$p1=plotmaple("Statistics[BoxPlot]([$A,$B,$C,
$D],outliers=true,deciles=false,
color=[blue,magenta,olive,red],transparency=0.5,orientation=horizontal,
tickmarks=[default,0]),
plotoptions='height=200, width=200'");

```

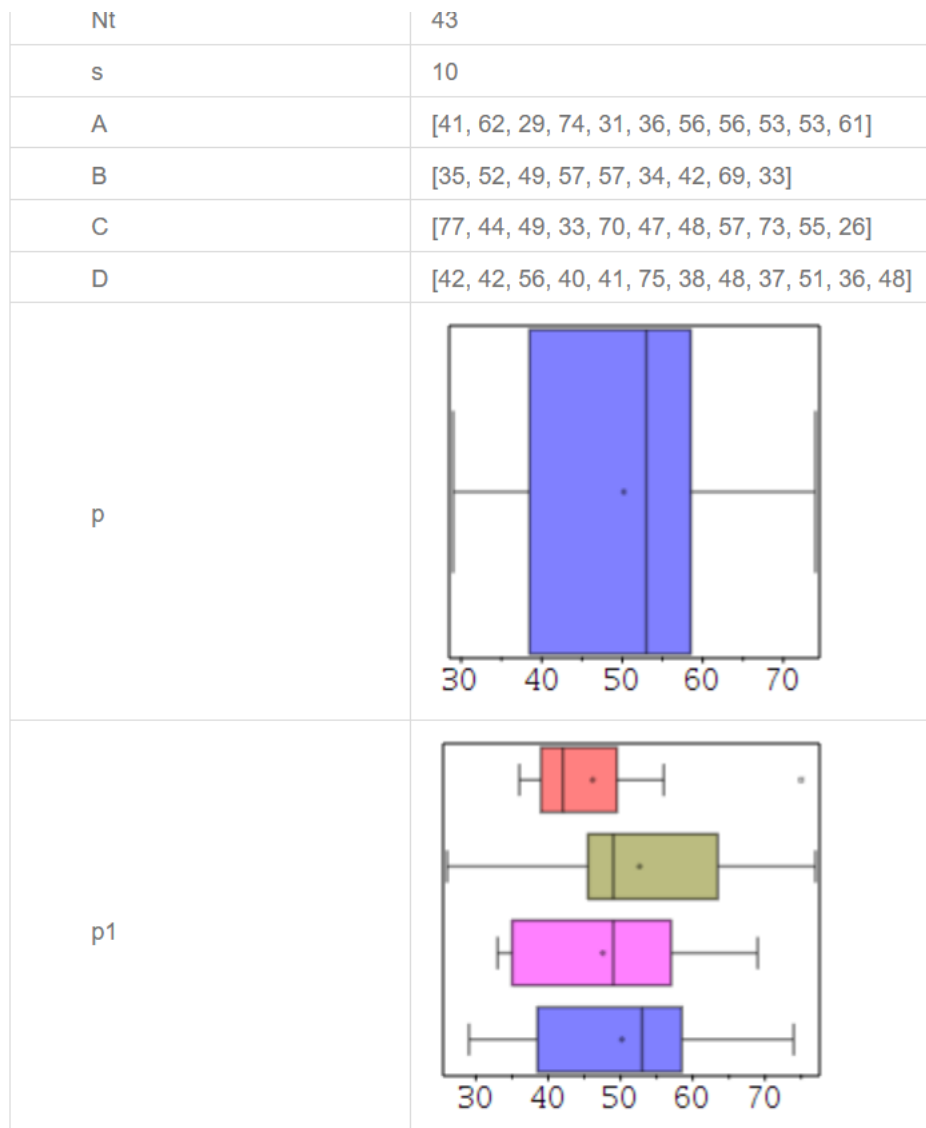


Figure 1.80: Boxplot

1.6.1.6 Scatterplot en trendlijn

In het volgende voorbeeld ziet u een dataset met X en met Y-waarden.

Door deze puntenwolk gaat een trendlijn waarvan de determinatiecoëfficiënt R^2 gevraagd wordt.

Preview

Gegeven zijn de gepaarde datasets X en Y .

Bereken met behulp van Excel het percentage van de variantie van Y dat door de onafhankelijke variabele x verklaard wordt.

x_i	y_i
1	11.1
2	12.6
4	13.8
5	14.1
3	13.1
2	13.3

TIP: Je kunt de tabel rechtstreeks met copy en paste in Excel plakken.

Het aandeel van de variantie van Y dat door x verklaard wordt is % (rond af op 2 decimalen)

Figure 1.81: Determinatiecoëfficiënt berekenen

Met twee datasets (van gelijke lengte) voor X - en Y -waarden is met het commando `Fit` uit het Statistics-pakket eenvoudig om de trendlijn op te vragen (hier een lineaire functie in x).

Met de extra optie `output=rsquared` krijgt u de determinatiecoëfficiënt R^2 .

De correlatiecoëfficiënt R is eventueel ook op te vragen met het commando `Correlation` uit dit Statistics-pakket.

(Wilt u de covariantie van de twee datasets weten, kan dit met `Covariance` uit het Statistics-pakket.)

De puntenwolk tekent u met het commando `ScatterPlot` uit het Statistics-pakket.

Het *Algorithm* voor trendlijn en determinatiecoëfficiënt is als volgt:

```
$x2=rand(2, 3, 1);
$x3=rand(3, 4, 1);
$x4=rand(4, 5, 1);
$x5=rand(1, 5, 1);
$x6=rand(1, 5, 1);
$y1=decimal(1, rand(10, 12));
$y2=decimal(1, rand(12, 13));
$y3=decimal(1, rand(13, 14));
$y4=decimal(1, rand(14, 15));
$y5=decimal(1, rand(9, 14));
$y6=decimal(1, rand(10, 15));
$fit=maple("Statistics[Fit](a+b*x, [$x1, $x2, $x3, $x4, $x5, $x6],
[$y1, $y2, $y3, $y4, $y5, $y6], x)");
$fitr=maple("Statistics[Fit](a+b*x, [$x1, $x2, $x3, $x4, $x5, $x6],
[$y1, $y2, $y3, $y4, $y5, $y6], x, output=rsquared)");
$Rsq=decimal(4, $fitr);
$cov=maple("Statistics[Covariance]([$x1, $x2, $x3, $x4, $x5, $x6],
[$y1, $y2, $y3, $y4, $y5, $y6])");
$cor=maple("Statistics[Correlation]([$x1, $x2, $x3, $x4, $x5, $x6],
[$y1, $y2, $y3, $y4, $y5, $y6])");
$R=sqrt($fitr);
$plot=plotmaple("plots[display]([Statistics[ScatterPlot]([$x1, $x2,
$x3, $x4, $x5, $x6],
[$y1, $y2, $y3, $y4, $y5, $y6], symbol=solidcircle, symbolsize=15),
```

```
plot($fit,x=0..6,color=red)],scaling=constrained,gridlines=true),
plotoptions='height=300, width=300');

```

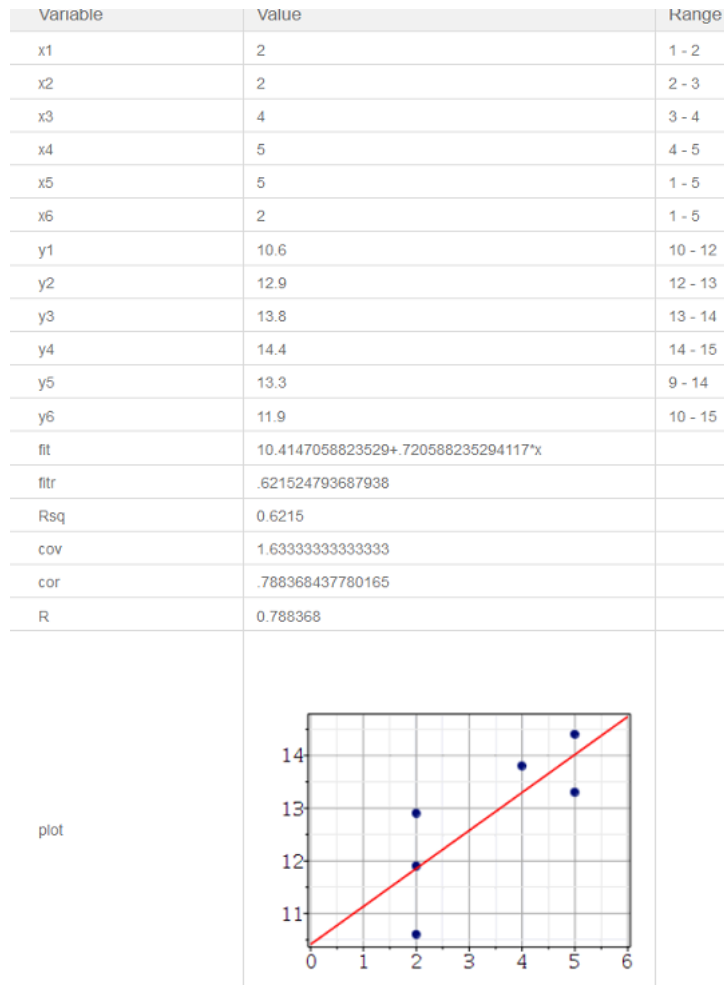


Figure 1.82: Trendlijn

1.6.1.7 Centrummaten

Hier ziet u een aantal Maple-commando's uit het Statistics-pakket om uitgaande van een dataset een aantal centrummaten te berekenen.

Als u de beschikking heeft over een frequentietabel waarbij de frequentie van elk gegeven vastligt, kunt u met dezelfde commando's deze centrummaten berekenen.

```
> data:=[1,2,3,4,5,6,7,8,9];
```

```
data := [1, 2, 3, 4, 5, 6, 7, 8, 9] (1.50)
```

```
> W:=[4,7,2,2,2,4,4,7,5]
```

```
W := [4, 7, 2, 2, 2, 4, 4, 7, 5] (1.51)
```

Gemiddelde waarde

De gemiddelde waarde van de dataset berekent u met het commando `Mean` uit het Statistics-pakket. Is van elk getal uit de dataset de frequentie gegeven door middel van de lijst `W`, dan is met de extra optie `weights=W` ook het gemiddelde te berekenen.


```
> Statistics[Mean](data);
```

5. (1.52)

```
> Statistics[Mean](data,weights=W);
```

5.27027027027027 (1.53)

Aantal gegevens

Met het commando `COUNT` uit het Statistics-pakket is het aantal van de dataset op te vragen. Als de frequentie van elke waarde uit de dataset gegeven is met de lijst `W` is duidelijk wat het totaal aantal gegevens is.

```
> Statistics[Count](data);
```

9 (1.54)

```
> Statistics[Count](data,weights=W);
```

37. (1.55)

De mediaan

De mediaan van een dataset vraagt u op met het commando `Median` uit het Statistics-pakket. Ook hier is de extra optie van de lijst `W` voor de frequenties van de data mogelijk.

```
> Statistics[Median](data);
```

5. (1.56)

```
> Statistics[Median](data,weights=W);
```

6. (1.57)

De kwartielen

Met het commando `Quartile` uit het Statistics-pakket is met de optie 1 het eerste kwartiel op te vragen van een dataset. Met de optie 2 en 3 krijgt u respectievelijk het tweede en derde kwartiel. Op dezelfde manier als de frequenties van de data gegeven zijn met de lijst `W`.

```
> Statistics[Quartile](data,1);
```

2.66666666666667 (1.58)

```
> Statistics[Quartile](data,1,weights=W);
```

1.99470899470899 (1.59)

De modus

De modus is de waarde uit de dataset die het vaakst voorkomt. Soms zijn er meer modi aan te wijzen. Ook hier kan de frequentie van alle data meegewogen worden met behulp van een lijst `W`.

```
> Statistics[Mode](data);
```

{1., 2., 3., 4., 5., 6., 7., 8., 9.} (1.60)

```
> Statistics[Mode](data,weights=W);
```

{2., 8.} (1.61)

In het laatste geval komen de waarden 2 en 8 het meest en evenveel voor. De frequentieverdeling is dus bimodaal.

1.6.1.8 Spreidingsmaten van een steekproef

Gegeven is weer een dataset als lijst en een lijst met frequenties.

```
> data:=[1,2,3,4,5,6,7,8,9];
```

```
data := [1, 2, 3, 4, 5, 6, 7, 8, 9] (1.62)
```

```
> W:=[4,7,2,2,2,4,4,7,5]
```

```
W := [4, 7, 2, 2, 2, 4, 4, 7, 5] (1.63)
```

De steekproefvariantie

De steekproefvariantie en de steekproef standaarddeviatie van de dataset bepaalt u met het commando `Variance` uit het Statistics-pakket als volgt:

```
> Statistics[Variance](data);
```

```
7.500000000000000 (1.64)
```

Bij een gegeven frequentieverdeling op dezelfde manier met de lijst van frequenties `W`.

```
> Statistics[Variance](data,weights=W);
```

```
9.27487352445194 (1.65)
```

De standaarddeviatie van de steekproef

```
> Statistics[StandardDeviation](data);
```

```
2.73861278752583 (1.66)
```

Bij een gegeven frequentieverdeling op dezelfde manier met de lijst van frequenties `W`.

```
> Statistics[StandardDeviation](data,weights=W);
```

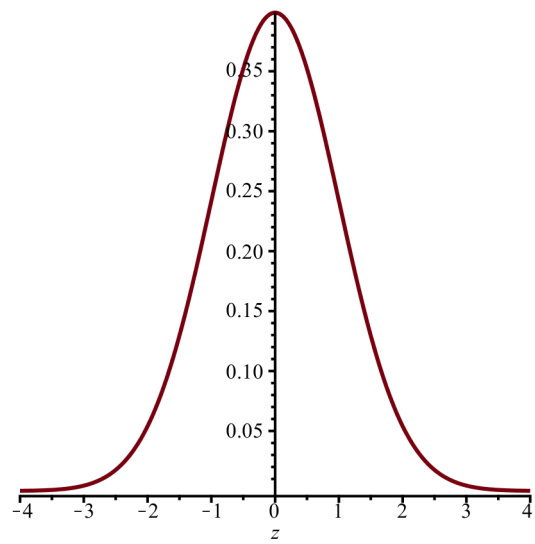
```
3.04546770208649 (1.67)
```

1.6.2 Kansverdelingen

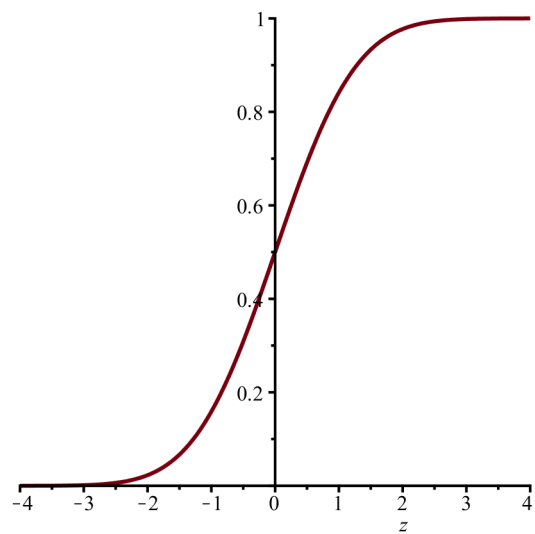
Bij verschillende kansdichtheidsfuncties (PDF = Probability Density Function) zijn cumulatieve kansen te berekenen met de cumulatieve kansdichtheidsfunctie (CDF = Cumulative Density Function) en de inversen daarvan (inverse cumulative distribution function = quantile function).

Hieronder ziet u hoe in Maple de drie functies in beeld gebracht zijn.

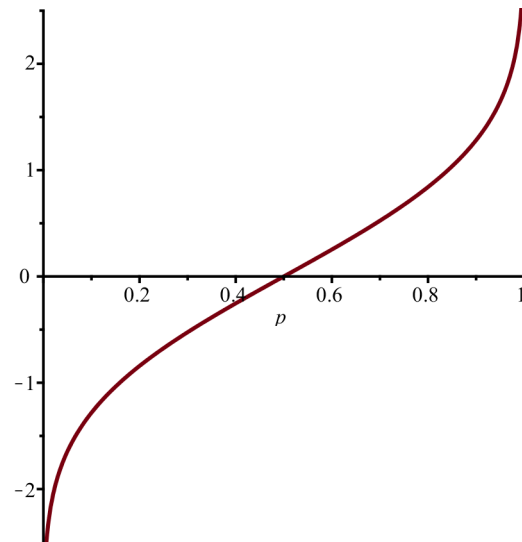
```
> plot(Statistics[PDF](Normal(0,1),z),z=-4..4);
```



```
> plot(Statistics[CDF](Normal(0,1),z),z=-4..4);
```



```
> plot(Statistics[Quantile](Normal(0,1),p),p=0..1);
```



TIP: De drie functies `Statistics[PDF]`, `Statistics[CDF]` en `Statistics[Quantile]` zijn te gebruiken voor allerlei kansverdelingen.

1.6.2.1 Standaardnormale verdeling

Cumulatieve kans

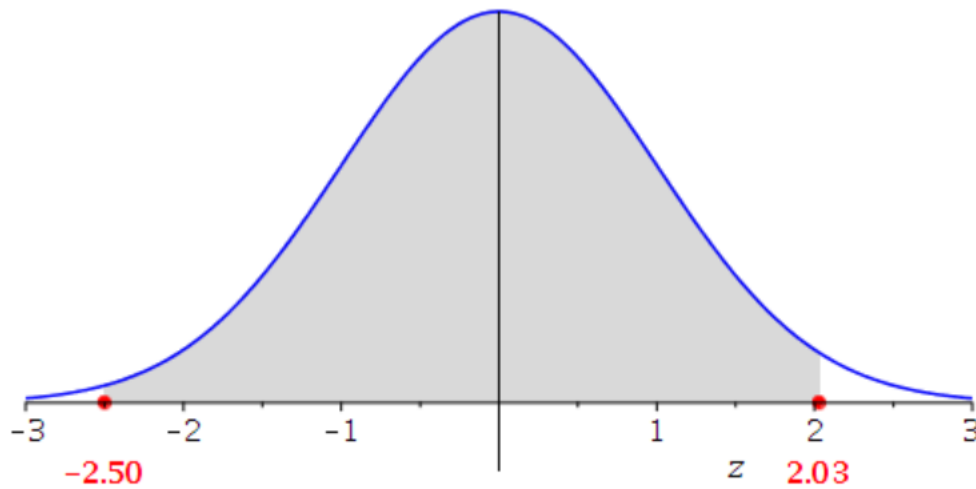
Voor de cumulatieve kansdichtheidsfunctie van de **standaardnormale verdeling** kunt u cumulatieve kansen berekenen $P(Z \leq z)$ met de functie `erf` in Möbius. Het is niet nodig om hierbij Maple in te schakelen.

De kansdichtheidsfunctie van de normaalverdeling met parameters μ en σ (PDF = Probability Density Function) is snel te genereren met het Maple-commando `Statistics[PDF](Normal($m, $s))`.

De kansvariabele Z volgt een standaardnormale verdeling.

In de grafiek zie je de kansdichtheid van deze standaard normale verdeling.

De oppervlakte onder de grafiek correspondeert met de kans $P(-2.50 \leq Z \leq 2.03)$.



Bereken deze kans die correspondeert met de gekleurde oppervlakte in de figuur.

(rond af op 4 decimalen)

Figure 1.83: Cumulatieve kans bij standaardnormale verdeling

Met het volgende *Algorithm* prepareert u de berekeningen en de dynamische figuur:

```
$a=numfmt("0.00", decimal(2, range(-2.8, 1.5, 0.01)));
$space=1.2;
$b=numfmt("0.00", decimal(2, range($a+$space, 2.8, 0.01)));
$pa=decimal(4, erf($a));
$pb=decimal(4, erf($b));
$opp=$pb-$pa;
$m=0;
$s=1;
$grafiek=plotmaple("
p0:=plot(Statistics[PDF](Normal($m,$s),x),x=($m)-3*$s..($m)+3*$s,
tickmarks=[default,[ ]],thickness=1,color=blue):
p1:=plot(Statistics[PDF](Normal($m,$s),x),x=$a..
$b,filled=true,color=gray):
p2 := plots[textplot]([($a), -0.07, $a],
color=red,font=[Times,Bold,12]):
p3 := plots[textplot]([($b), -0.07, $b],
color=red,font=[Times,Bold,12]):
p4:=plots[pointplot]([[$a,0],
[$b,0]],color=red,symbol=solidcircle,symbolsize=15):
plots[display]({p0,p1,p2,p3,p4},labels=[z,``]),
plotoptions='height=250, width=500' ");
```

TIP: De cumulatieve kans $P(z \leq $a)$ voor de normale verdeling is ook te berekenen met Maple:

```
$pa=maple("Statistics[CDF](Normal($m,$s),$a)");
```

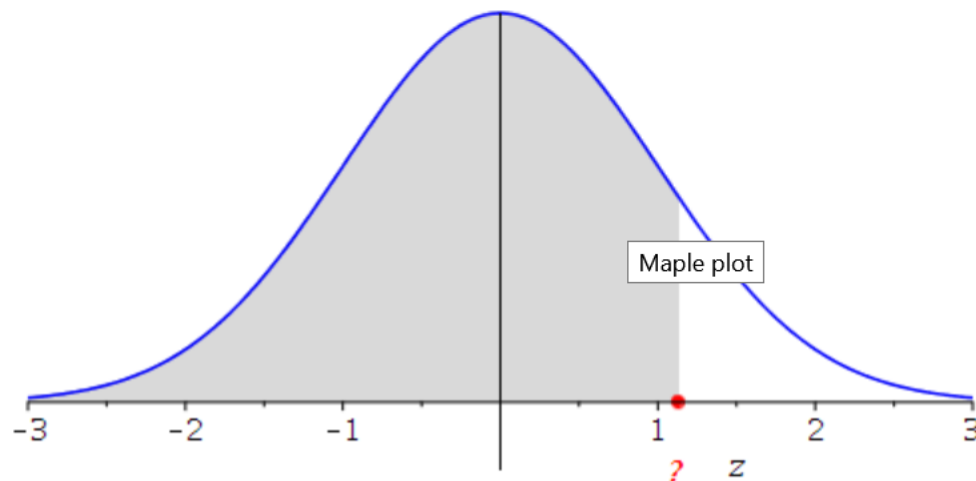
Inverse cumulatieve kansdichtheidsfunctie

Met de inverse van de cumulatieve kansdichtheidsfunctie (Quantile) is de grenswaarde te berekenen.

De kansvariabele Z volgt een standaardnormale verdeling.

In de grafiek zie je de kansdichtheid van deze standaard normale verdeling.

De oppervlakte onder de grafiek correspondeert met de kans $P(Z \leq ?) = 0.87$.



Bereken de waarde van Z op de plaats van het vraagteken.

Number (rond af op 2 decimalen)

Figure 1.84: Inverse cumulatieve kans bij standaardnormale verdeling

Voor de inverse van de standaardnormale kansdichtheidsfunctie berekent u met de Möbiusfunctie `inverf` de grenswaarde z bij een cumulatieve kans P die correspondeert met een oppervlakte onder de kromme van deze kansdichtheidsfunctie.

Het is dus niet nodig om hiervoor een Maple commando te gebruiken. U ziet het in het volgende *Algorithm*.

```
$m=0;
$s=1;
$opp=switch(rint(2),rand(.1,.4,2),rand(.6,.9,2));
$z=decimal(2,inverf($opp));
$grafiek=plotmaple("
p0:=plot(Statistics[PDF](Normal($m,$s),x),x=($m)-3*$s..($m)+3*$s,
tickmarks=[default,[ ]], thickness=1,color=blue):
p1:=plot(Statistics[PDF](Normal($m,$s),x),x=($m)-3*$s..$z,
filled=true,color=gray):
p2 := plots[textplot]([($z), -0.07, `?`],
color=red,font=[Times,Bold,12]):
p3:=plots[pointplot]
([$z,0],color=red,symbol=solidcircle,symbolsize=15):
plots[display]({p0,p1,p2,p3},labels=[z,``]),
plotoptions='height=250, width=500' ");
```

TIP: De grenswaarde z waarvoor geldt dat $P(Z \leq z) = \$opp$ voor de normale verdeling is ook te berekenen met Maple:

```
$z:=maple("Statistics[Quantile](Normal($m,$s),$opp)");
```

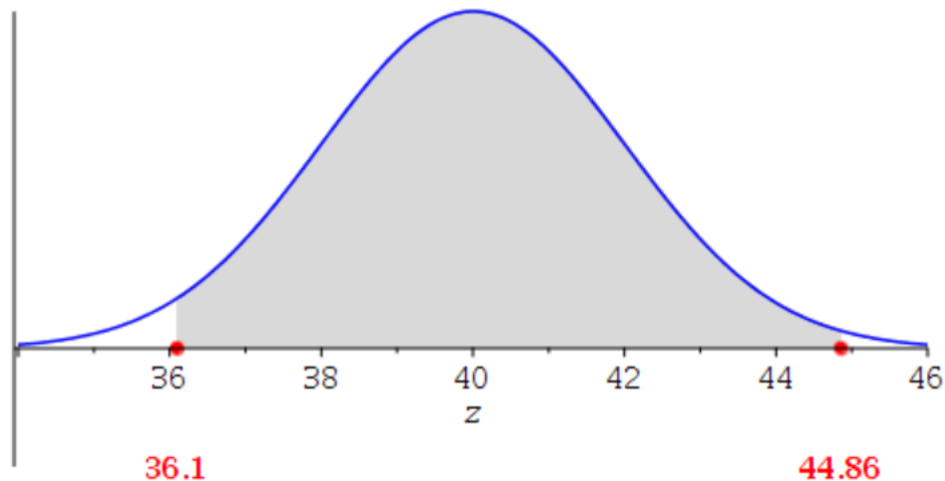
1.6.2.2 Normale verdeling algemeen

In het volgende voorbeeld is de grafische presentatie van de kans te zien waarbij de normale verdeling een rol speelt.

De kansvariabele X volgt een normale verdeling met $\mu = 40$ en $\sigma = 2$.

In de grafiek zie je de kansdichtheid van deze normale verdeling.

De oppervlakte onder de grafiek correspondeert met de kans $P(36.1 \leq X \leq 44.86)$.



Bereken deze kans die correspondeert met de gekleurde oppervlakte in de figuur.

(rond af op 4 decimalen)

Figure 1.85: Normale verdeling

Het bijbehorende *Algorithm* is als volgt:

```
$m:=range(10, 40, 5);
$s:=range(2, 5);
$z1:=decimal(2, range(-2.8, 1.5, 0.01));
$space=1.2;
$z2:=decimal(2, range($z1+$space, 2.8, 0.01));
$a=$m+($z1)*$s;
$b=$m+($z2)*$s;
$Opp:=maple("Statistics[CDF](Normal($m,$s),$b)-Statistics[CDF]
(Normal($m,$s),$a)");
$opp:=decimal(4, $Opp);
$grafiek:=plotmaple("
p0:=plot(1/($s*sqrt(2*Pi))*exp(-1/2*((x-($m))/($s))^2), x=($m)-3*$s..
($m)+3*$s, tickmarks=[default, []], thickness=1, color=blue):
p1:=plot(1/($s*sqrt(2*Pi))*exp(-1/2*((x-($m))/($s))^2), x=$a..
$b, filled=true, color=gray):
```

```
p2 := plots[textplot]([($a), -0.07, $a],  
  color=red,font=[Times,Bold,12]):  
p3 := plots[textplot]([($b), -0.07, $b],  
  color=red,font=[Times,Bold,12]):  
p4:=plots[pointplot]([[$a,0],  
  [$b,0]],color=red,symbol=solidcircle,symbolsize=15):  
plots[display]({p0,p1,p2,p3,p4},labels=[z,``]),  
plotoptions='height=250, width=500' ");
```


Index

