

Naam: Klas: Studentnummer:

Instructie

Voor je ligt de herkansing van CPL01. Bij deze toets moeten enkele vragen worden uitgewerkt in Code::Blocks. De toets neemt 90 minuten in beslag. Tijdens de toets mag je het boek “C How to Program” gebruiken.

Internet, andere documenten en andere programma's dan Code::Blocks zijn niet toegestaan en zullen worden opgemerkt door de CPL01 logger.

De bestanden `jouwklas_jouwnaam_jouwstudnr_vraag1.c` en `jouwklas_jouwnaam_jouwstudnr_vraag2.c` zullen na het afronden van de opdrachten moeten worden ingeleverd in de inlevermap van CPL01. Dit kan je doen in de map:
`..\inlever\ELEKTROTECHNIEK\2015-2016\CPL01\Herkansing`

Na het inleveren van je `.c` bestanden dien je dit opgaveblad, met je naam, klas en studentnummer ingevuld, terug te geven aan de docent. Zonder ingeleverd opgaveblad wordt je code niet nagekeken!

Indien je vragen of problemen hebt die niet aan de programmeeropdracht zijn gerelateerd is de docent bereid te helpen. Succes met de toets!

Toetsvraag 1, 45 punten

Schrijf een functie genaamd `sni_jpuntLijnen` die het snijpunt van twee lijnen berekent. De lijnen zijn beschreven met de volgende formules:

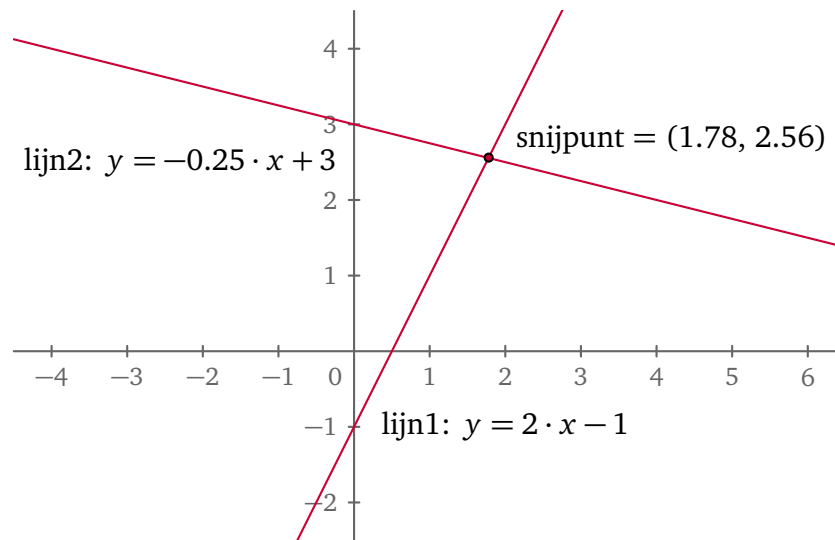
$$y = a_1 \cdot x + b_1 \quad \text{en} \quad y = a_2 \cdot x + b_2 \quad (1)$$

Twee lijnen kunnen géén, één of oneindig veel snijpunten hebben.

- Als a_1 gelijk is aan a_2 maar b_1 ongelijk is aan b_2 , dan lopen de lijnen parallel (en is er geen snijpunt).
- Als a_1 gelijk is aan a_2 en bovendien b_1 gelijk is aan b_2 , dan zijn de lijnen gelijk (en zijn er oneindig veel snijpunten).
- Als a_1 ongelijk is aan a_2 , dan is er één snijpunt. De x - en y -coördinaat van dit snijpunt kunnen dan berekend worden met vergelijking (2). Een voorbeeld van twee lijnen met één snijpunt is gegeven in figuur 1.

$$x = \frac{b_2 - b_1}{a_1 - a_2} \quad \text{en} \quad y = a_1 \cdot x + b_1 \quad (2)$$

In listing 1 is een onvolledig C programma gegeven waar alleen de functie `sni_jpuntLijnen` nog moet worden ingevuld en aangeroepen. Deze code kun je kopiëren naar je eigen pro-



Figuur 1: Het snijpunt van twee lijnen.

programma vanuit het bestand:

```
... \opleiding\Elektrotechniek\2015-2016\CPL01\Toetsen\Herkansing\vraag1.c
```

De functie `snijpuntLijnen` heeft zes parameters: vier invoerparameters (call by value) waarmee de parameters a_1 , b_1 , a_2 en b_2 aan de functie worden doorgegeven en twee uitvoerparameters (call by reference) waarmee de coördinaten x en y van het snijpunt vanuit de functie worden teruggegeven. De functie moet de integer waarde 1 teruggeven als er één snijpunt is. In dit geval moeten de x - en y -coördinaat van dit snijpunt berekend en teruggegeven worden via de twee uitvoerparameters. De functie moet de waarde 0 teruggeven als er géén snijpunt is en 2 als er oneindig veel snijpunten zijn.

Vul de code en de aanroep van de functie `snijpuntLijnen` in zodat het programma dat gegeven is in listing 1 naar behoren werkt. Als voorbeeld zijn twee runs van het programma gegeven in figuur 2 (de door de gebruiker ingevoerde getallen zijn onderstreept weergegeven).

Geef a1: 2

Geef b1: -1

Geef a2: -0.25

Geef b2: 3

Snijpunt van lijnen $y = 2.00x + -1.00$ en $y = -0.25x + 3.00$ is (1.78, 2.56)

Geef a1: 2

Geef b1: -1

Geef a2: 2

Geef b2: 3

De lijnen $y = 2.00x + -1.00$ en $y = 2.00x + 3.00$ lopen parallel

Figuur 2: Gewenste uitvoer van het programma van toetsvraag 1.

```
1 #include <stdio.h>
2
3 int snijpuntLijnen(double a1, double b1, double a2, double b2, ←
   ↪ double *x, double *y)
4 {
5     /* Vul hier je code in */
6 }
7
8 double scanDouble(char* text)
9 {
10    double d;
11    do
12    {
13        printf("%s: ", text);
14        fflush(stdin);
15    }
16    while (scanf("%lf", &d) != 1);
17    return d;
18 }
19
20 int main(void)
21 {
22    double a1 = scanDouble("Geef a1");
23    double b1 = scanDouble("Geef b1");
24    double a2 = scanDouble("Geef a2");
25    double b2 = scanDouble("Geef b2");
26
27    double x, y;
28    int res = /* Roep hier de functie snijpuntLijnen aan */;
29    if (res == 1)
30    {
31        printf("Snijpunt van lijnen  $y = 0.2f x + 0.2f$  en  $y = 0.2f x + 0.2f$  ←
   ↪  $x + 0.2f$  is  $(0.2f, 0.2f)$ ", a1, b1, a2, b2, x, y);
32    }
33    else
34    {
35        if (res == 0)
36        {
37            printf("De lijnen  $y = 0.2f x + 0.2f$  en  $y = 0.2f x + 0.2f$  ←
   ↪  $x + 0.2f$  lopen parallel", a1, b1, a2, b2);
38        }
39        else
40        {
41            printf("De lijnen  $y = 0.2f x + 0.2f$  en  $y = 0.2f x + 0.2f$  ←
   ↪  $x + 0.2f$  zijn gelijk", a1, b1, a2, b2);
42        }
43    }
44
45    return 0;
46 }
```

Listing 1: Een onvolledig C-programma behorende bij vraag 1.

Toetsvraag 2

In het bestand:

```
...\Opleiding\Elektrotechniek\2015-2016\CPL01\Toetsen\Deeltoets2\vraag2.c
```

is een onvolledig C programma gegeven. Zoals je ziet is in dit programma een 2-dimensionale array met integers gedefinieerd genaamd `plaatje` die bestaat uit 18 rijen van elk 18 kolommen.

Toetsvraag 2, deel A, 20 punten

Geef de implementatie van de functie `print` die de inhoud van de 2-dimensionale `int`-array genaamd `p` die bestaat uit 18 rijen van elk 18 kolommen, rij voor rij afdruckt op het beeldscherm. Het prototype van de functie is als volgt:

```
void print(int p[18][18], char char0, char char1);
```

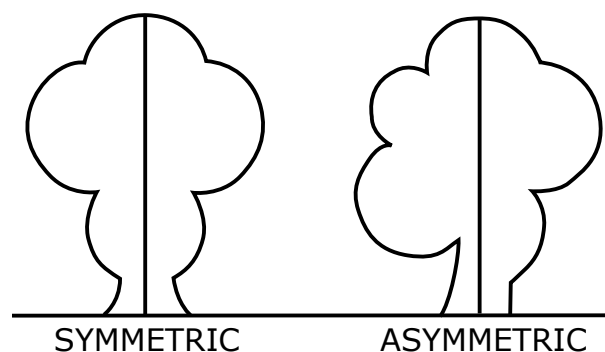
Voor elke 0 in `p` moet het als tweede argument meegegeven karakter worden afgedrukt en voor elke 1 in `p` moet het als derde argument meegegeven karakter worden afgedrukt.

Als het goed is, zie je een plaatje van een panda.

Toetsvraag 2, deel B, 25 punten

Geef de implementatie van de functie `isSymmetrischInYAs` die bepaalt of de 2-dimensionale `int`-array genaamd `p` symmetrisch is in de Y-as. Als dit zo is, dan moet de functie de waarde 1 teruggeven maar als dit niet zo is, dan moet de functie de waarde 0 teruggeven.

We gaan er hierbij vanuit dat de X- en Y-as het plaatje in 4 gelijke delen verdelen. Een plaatje is dan symmetrisch in de Y-as als het plaatje niet verandert als we het spiegelen in de Y-as. In figuur 3 staan twee plaatjes van bomen afgebeeld. Het linker plaatje is symmetrisch in de Y-as maar het rechter plaatje niet.



Figuur 3: Een symmetrische en een niet symmetrische boom (bron: Wikipedia).