

1. INHOUD

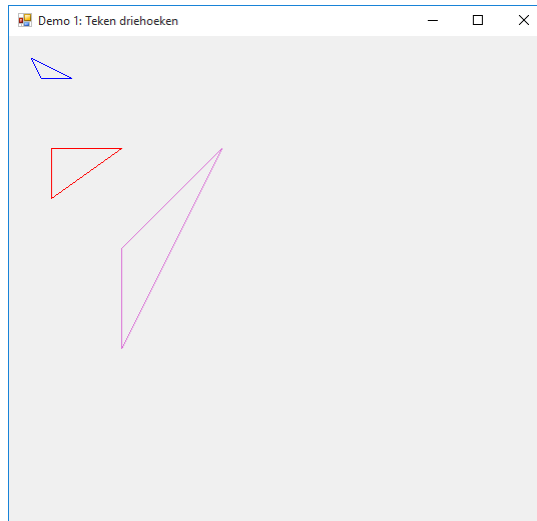
Procedures (functies zonder return-waarde)

2. OEFENINGEN

- Demo 1: Teken driehoeken
- Demo 2: Print array of double
- A: Stapel blokken
- A: Weerstanden 1
- A: Weerstanden 2
- A: Draw Polygon
- A: Biljarttafel
- E: Bijenkorf
- X: Slinger

2.1 Demo 1: Teken driehoeken

Schrijf een programma dat verschillende driehoeken kan tekenen. Om het tekenen van deze driehoeken maar één keer te moeten schrijven, wordt het gebruik van een procedure aangetoond. Deze procedure neemt de hoekpunten van de driehoek op als argumenten tesamen met de gewenste kleur.



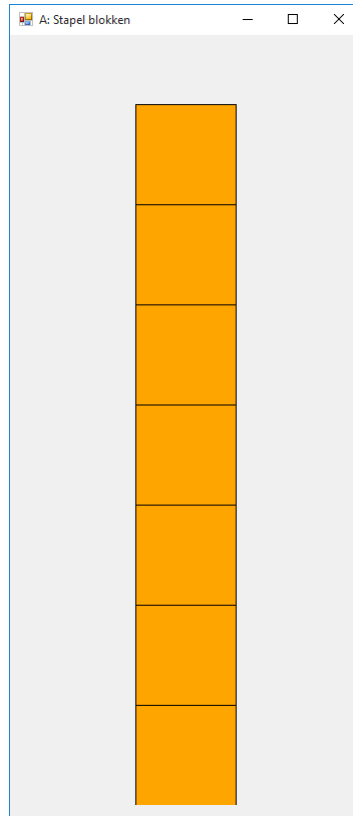
Figuur 1: Voorbeeld oplossing: Demo 1

2.2 Demo 2: Print array of double

Het uitprinten van een array in een label kan gebeuren met een for-loop. Echter kan het uitprinten van een array plaatsvinden in een procedure. Schrijf een procedure die toelaat om een array van doubles te printen in een label.

2.3 A: Stapel blokken

Schrijf een procedure die toelaat om een vierkant te tekenen. Dit vierkant heeft een grootte van 25 bij 25 pixels. De procedure neemt het middelpunt van het vierkant via X en Y als argumenten. Stapel 7 van deze blokken op door de functie binnen een for-loop aan te roepen. Pas de argumenten van de functie telkens aan zodat een toren bekomen wordt.



Figuur 2: Voorbeeld oplossing: Stapel blokken

2.4 A: Weerstanden 1

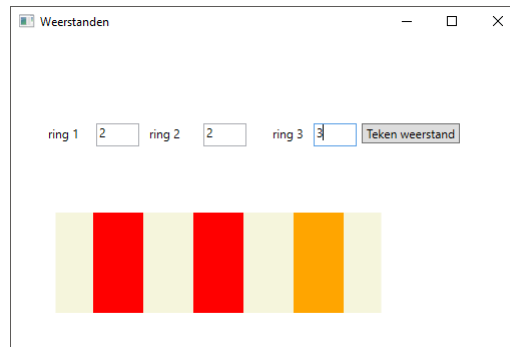
Schrijf een procedure die toelaat om a.d.h.v. 3 getallen van 0 t.e.m. 9 de waarde van de weerstand te bepalen. De getallen komen overeen met de ringen die te vinden zijn op echte weerstanden. Hieronder worden een aantal voorbeelden weergegeven.

Ring 1	Ring 2	Ring 3	Waarde
1	5	2	1.5 k Ω
2	2	3	33 k Ω
3	9	3	39 k Ω
1	0	5	1.0 M Ω
1	2	0	12 Ω

2.5 A: Weerstanden 2

Deze opgave is heel gelijkaardig aan voorgaande opgave. Laat de gebruiker toe om 3 getallen in te geven zoals in de voorgaande opgave. A.d.h.v. deze 3 ingegeven waarden worden de ringen van de weerstand in een canvas getekend. Hiervoor kunnen rechthoeken gebruikt worden om de ringen te tekenen. De weerstand wordt getekend door een lange beige gevulde rechthoek. Elke ring is 50 pixels breed en 100 pixels hoog. De beige rechthoek is 350 pixels breed en ook 100 pixels hoog.

Hint: Gebruik een procedure om elk van de ringen te tekenen. Deze procedure heeft als argumenten: de horizontale positie van de ring en de kleur van de ring.



Figuur 3: Voorbeeldprogramma weerstanden

2.6 A: Draw Polygon

Schrijf een programma met daarin een procedure die een polygon kan tekenen. De polygon bestaat uit een aantal punten die door lijnen verbonden worden. Als argumenten neemt de procedure 2 arrays en een kleur aan. De 1^{te} array bevat alle x-coördinaten terwijl de 2^{de} array de y-coördinaten bevat. Beide arrays zijn uiteraard even lang! De coördinaten mogen vast in het programma opgegeven zijn.

2.7 A: Biljarttafel

Schrijf een procedure die een gevulde schijf kan tekenen. Deze procedure neemt het middelpunt, straal en de kleur van de bol als argumenten op. Via een timer teken je 2 bollen die over een canvas rollen. Zorg ervoor dat deze bollen tegen de wanden van de panel kunnen weerkaatsen.

Hint: Gebruik een timer om een vloeiende beweging tot stand te brengen.

Hint: Gebruik global variabelen om de positie van de bollen bij te houden (X en Y). Hetzelfde gaat op voor de snelheid van beide bollen (v_x en v_y).

2.8 E: Bijenkorf

Schrijf een procedure die een zeshoek kan tekenen. Deze procedure heeft als argumenten:

- x coördinaat,
- y coördinaat,
- de straal van de zeshoek.

Een zeshoek bestaat uit 6 punten die evenver liggen van het middelpunt.

Hint: Een zeshoek heeft 6 gelijke zijden die elk overeenkomen met een hoekverdraaiing van $\frac{2 \cdot \pi}{6}$.

Roep deze functie aan om verschillende zeshoeken naast elkaar te tekenen. Op deze manier kan je een bijenkorf nabootsen.



Figuur 4: Voorbeeld oplossing: Bijenkorf

2.9 X: Slinger

Teken en simuleer de beweging van een massa aan een slinger. De bewegingsvergelijking (vgl. 1) wordt gedefinieerd aan de hand van de hoekversnelling $\frac{d^2\theta}{dt^2}$, de hoek θ , de masse m en de lengte l . θ is de hoek tussen de slinger en de verticale as, m de massa onderaan de slinger en l de lengte van de slinger.

$$m * l * \ddot{\theta} = -m * g * \sin(\theta) \quad (1)$$

Door vergelijking 2 te herleiden naar de versnelling $\frac{d^2\theta}{dt^2}$ kan de versnelling van de massa voor elke hoek bepaald worden. De versnelling vermenigvuldigen met de tijd geeft ons de snelheid v van de massa. De snelheidsvergelijking (vgl. 2) wordt dan:

$$\dot{\theta}_t = \dot{\theta}_{t-1} + \frac{g * \sin(\theta)}{l} * dt \quad (2)$$

De x- en y-positie van de slinger (eindpunt) worden dan bepaald volgens:

$$\begin{aligned} x &= l \cdot \cos\left(\theta + \frac{\pi}{2}\right) \\ y &= l \cdot \sin\left(\theta + \frac{\pi}{2}\right) \end{aligned} \quad (3)$$

Veronderstel dat de slinger beweegt zonder wrijving en volledig stil hangt als hij volledig horizontaal hangt.