

# Objectgeoriënteerd Programmeren: WPO 1

---

## 1. INHOUD

Oprissing syntax, programmeermethodes, datatypes, functies/procedures, tekenen in C#. Herhaling Informatica 1<sup>ste</sup> bachelor.

## 2. OEFENINGEN

- Demo 1: Volume bol
- Demo 2: Tel totdat...
- Demo 3: Geometrische figuren
- Demo 4: Vul de 1D-array/lijst
- Demo 5: Swapfunctie
- A: Fahrenheit to Celsius
- A: Binnen bereik
- A: Why do I print?
- A: Schaakbord
- A: Kogelwerpen
- A: Biljarttafel
- A: Interpoleer
- A: TryParse(double)
- E: Maaltafels
- E: Bijenkorf
- E: Waveforms

### 2.1 Demo 1: Volume bol

Bereken het volume van een bol. Het volume van een bol wordt gegeven door vergelijking 1.

$$V_{bol} = \frac{4\pi r^3}{3} \quad (1)$$

Schrijf een programma dat de straal van de bol via een tekstveld inleest, en het resultaat in een ander tekstveld weergeeft. Voozie eveneens ook een knop “Bereken” om de berekening uit te voeren.

## 2.2 Demo 2: Tel totdat...

Schrijf een programma dat de gebruiker om 2 getallen vraagt. Tel vanaf het eerste getal af totdat het 2<sup>de</sup> getal bereikt is. Doe dit eerst a.d.h.v. een for-loop. Bouw daarna de loop om tot een while-loop.

## 2.3 Demo 3: Geometrische figuren

In deze demo wordt het tekenen van eenvoudige geometrische figuren getoond. Om te tekenen wordt er een canvas op het formulier geplaatst. Verander de achtergrondskleur naar de gewenste kleur (hier zwart). Teken een horizontale witte lijn, waarop een rechthoek wordt getekend. Hiernaast wordt ook een rode bol getekend.

**Hint 1:** Het assenkruis om te tekenen (canvas) is als volgt:

- het nulpunt van de tekening bevindt zich linksboven,
- horizontaal loopt de x-as van links naar rechts,
- verticaal loopt de y-as van boven naar beneden, dit betekent dat er in de verticale richting omgekeerd getekend wordt (spiegelen).

Om een tekening volgens de y-as te plotten, vindt volgende omzetting plaats:

$$y_{plot} = Height_{canvas} - y \quad (2)$$

## 2.4 Demo 4: Vul de 1D-array/lijst

Schrijf een programma waarin een 1D-array gebruikt wordt. Deze array wordt opeenvolgend gevuld door gebruik te maken van een loop. Nadien worden de waarden van deze array geprint binnen een tekstbox door gebruik te maken van een andere loop. Herhaal deze opgave door gebruik te maken van een lijst.

## 2.5 Demo 5: Swapfunctie

In dit programma wordt een swapfunctie geschreven die toelaat om 2 getallen van klein naar groot te sorteren indien nodig. Laat deze los op een array van ongesorteerde getallen. Print zowel de originele array als het resultaat na sorteren af. De swapfunctie wordt hierbij opgeroepen met 2 argumenten die “by reference” worden doorgegeven.

## 2.6 A: Fahrenheit to Celsius

Een weerstation beschikt over thermometers die enkel in Fahrenheit (F°) geijkt zijn. Om dezelfde temperatuur in graden Celcius (C°) te bekomen, moet volgende omzetting plaatsvinden:

$$T_{Celcius} = \frac{5}{9}(T_{Fahrenheit} - 32) \quad (3)$$

Schrijf een programma dat de temperatuur in Fahrenheit inleest, en de temperatuur in Celcius weergeeft. Bij het weergeven dient ook de eenheid vermeld te worden! De output wordt in een label weergegeven.

## 2.7 A: Binnen bereik

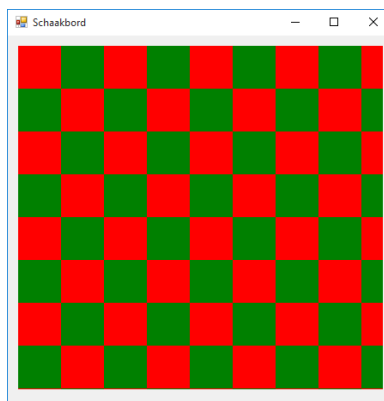
Schrijf een programma dat een getal van de gebruiker opvraagt. Via de knop “Vergelijk” wordt nagegaan of deze waarde zich tussen -100 en +100 bevindt. Indien dit zo is, wordt “Binnen bereik” in een messagebox getoond. In het andere geval geeft de messagebox “Too bad, retry!!!” weer.

## 2.8 A: Why do I print?

In deze opgave wordt er gevraagd om de getallen van 0 t.e.m. X (door de gebruiker in te geven) te printen. Als het getal een veelvoud van 8 is, wordt er ‘X’ geprint. Is het getal een veelvoud van zowel 8 en 5, wordt er ‘Y’ geprint.

## 2.9 A: Schaakbord

Teken een schaakbord door gebruik te maken van rode en groene vierkantjes die elkaar afwisselen (figuur 1). Elk vierkantje is 50 bij 50 pixels groot.



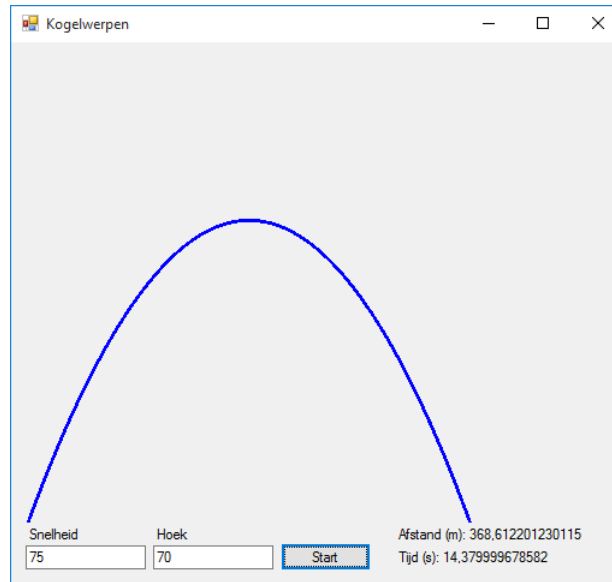
Figuur 1: Voorbeeld programma

## 2.10 A: Kogelwerpen

In een partijtje kogelwerpen worden massa's geworpen. De kromme die de massa's volgen wordt beschreven als:

$$\begin{cases} x(t) = v \cdot \cos(\alpha) \cdot t \\ y(t) = v \cdot \sin(\alpha) \cdot t - \frac{1}{2} \cdot g \cdot t^2 \end{cases} \quad (4)$$

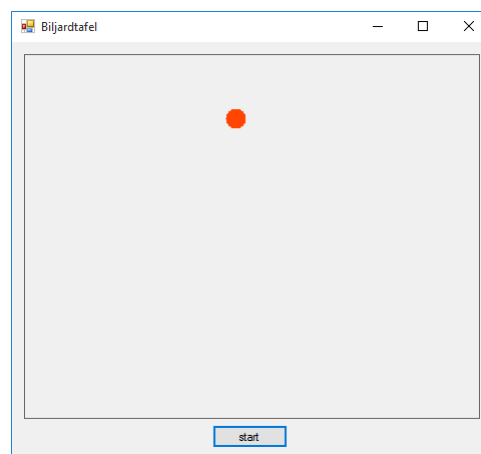
Plot de baan die deze kogel aflegt. De beginsnelheid en de afschiethoek worden ingegeven worden door de gebruiker. Let hierbij op dat het mogelijk is dat de plot niet op het scherm past. Wanneer en op welke afstand raakt de kogel de grond? Ga ervan uit dat de kogel vanaf de grond afgeschoten wordt, en maak in het programma gebruik van een do-while lus.



Figuur 2: Voorbeeld kogelwerpen

## 2.11 A: Biljarttafel

Schrijf een programma dat een bal op een veld laat rollen. Zorg ervoor dat deze bal bij het botsen tegen een wanden terugkaatst. Stel de wanden van het veld gelijk aan de randen van de canvas waarop je tekent. De beweging van deze bal wordt voorgesteld door een X en Y component. Samen vormen deze componenten een snelheidsvector. Dankzij een timer kan de beweging van de bal gevisualiseerd worden (zet deze aan, met een interval tussen 10 ms en 250 ms). De X en Y beweging worden aangepast in functie van de timerfrequentie, zodat deze bal een al dan niet vloeiende beweging uitvoert. De terugkaatsing kan plaatsvinden door één element van de snelheidsvector van teken te doen wisselen. Als de bal tegen een verticale wand aanbotst, wordt de horizontale richting omgedraaid. In het het geval dat de bal tegen een horizontale wand aanbotst, wordt de verticale richting omgedraaid (tekenwissel). Een voorbeeld van deze applicatie kan teruggevonden worden in figuur 3.



Figuur 3: Voorbeeld biljarttafel

## 2.12 A: Interpoleer

In onderstaande tabel worden alle getallen gegeven die op een even index staan. Het is nu aan jou om de waarden op de oneven plaatsen te berekenen. Die zijn momenteel aangeduid met “X”. Dit doe je door het gemiddelde te nemen van de naburige elementen. Print de hele array af in een tekstbox. Op hoeveel elementen moet je de array voorzien?

Index	Waarde
0	10
1	X
2	15
3	X
4	18
5	X
6	10
7	X
8	-10
9	X
10	-50

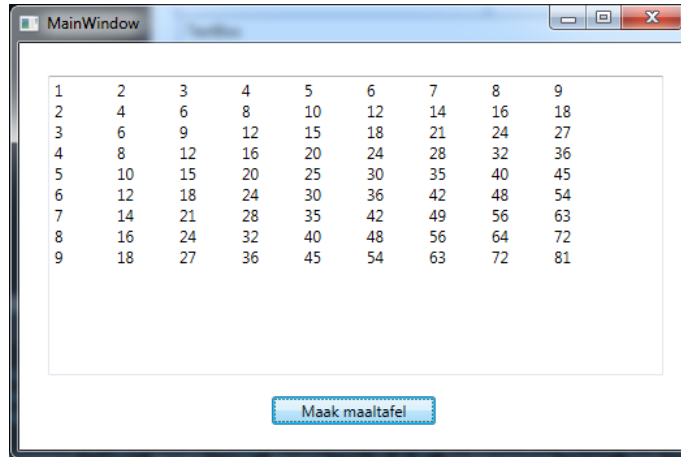
Kan je deze oefening herhalen door gebruik te maken van een list?

## 2.13 A: TryParse(double)

De klassieke “double.Parse” conversie laat toe om string naar een kommagetal om te zetten. Echter kan het zijn dat een gebruiker een gewoon karakter ingeeft (anders dan de punt of komma) waardoor deze omzettingmethode niet goed zal functioneren. Schrijf een functie die eerst nagaat of de ingegeven tekst effectief enkel een getal bevat. Indien dit zo is wordt deze tekst omgezet in een getal. De functie retourneert een boolean waarde (true indien gelukt, false indien niet gelukt). De waarde na omzetting (double) wordt via de “out”-statement aan de oproepende functie teruggestuurd.

## 2.14 E: Maaltafels

In de lagere school zal je ongetwijfeld de maaltafels hebben gezien. In deze opgave zal je een dergelijke tafel programmeren. Zorg ervoor dat je via 2 geneste for-loops onderstaande output (zie figuur 4) (textbox/label) kan genereren.



Figuur 4: Voorbeeldprogramma maaltafels.

**Hint:** Gebruik “\n” aan elk regeleinde.

## 2.15 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 elk 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.



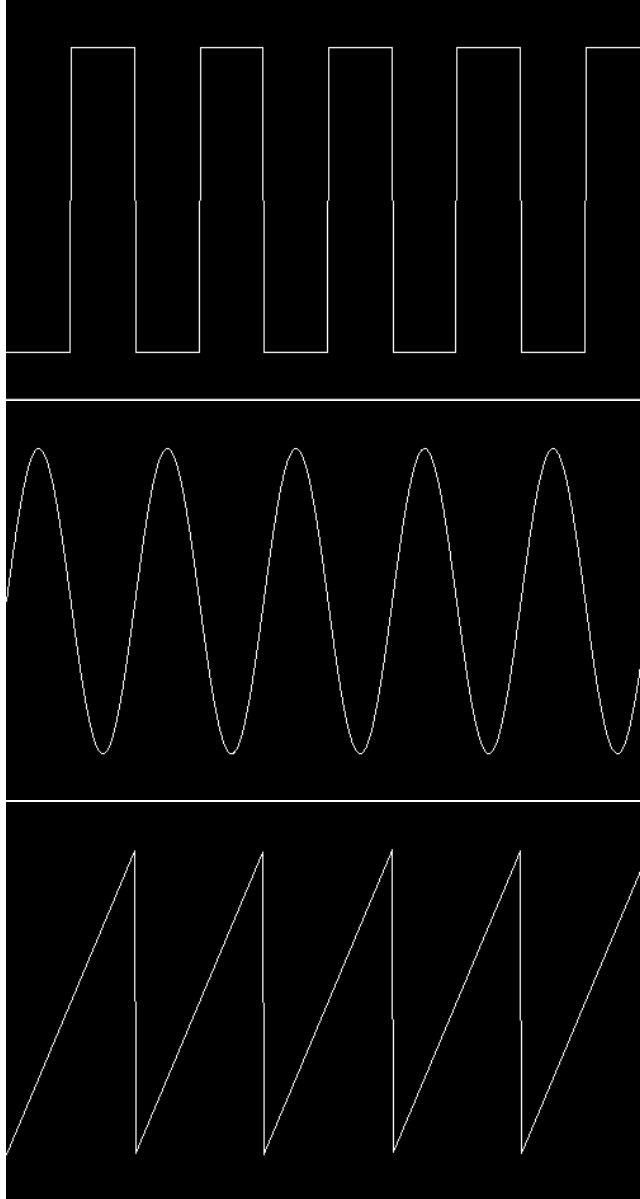
Figuur 5: Voorbeeld oplossing: Bijenkorf

## 2.16 E: Waveforms

Schrijf een programma waarin je 4 functies voorziet. De eerste 3 functies berekenen een geluidsgolf (zie hieronder voor de type geluidsgolven) en retourneren een array met daarin de wave. De 4<sup>de</sup> functie tekent een geluidsgolf a.d.h.v. een array van waarden. De te tekenen geluidsgolven zijn:

- een blokgolf,
- een zaagtand,
- een sinusoidaal.

Deze worden weergegeven in afbeelding 6.



Figuur 6: Voorbeelden van een blokgolf, zaagtand en sinusgolf.

Elk van deze functies retourneert een array met daarin de samples. Elke functie heeft volgende argumenten:

- de breedte van de canvas (pixels),
- de frequentie van de golf: aantal keren op en afdalen van de golf t.o.v. de breedte van de canvas,
- de amplitude: verhouding t.o.v. de hoogte van de canvas (een amplitude van 1 neemt dus de volledige canvashoogte in).

De 4<sup>de</sup> en laatste functie tekent 1 golf uit op de canvas. De keuze van welke golf getekend wordt, wordt bepaald door de keuze van een combobox. Deze functie heeft als argument de



array met de samples en retourneert niets (tekent alleen). In het programma zijn de frequentie, golfvorm en amplitude in te geven door de gebruiker.