

## 1. INHOUD

Functies (functies met return-waarde)

## 2. OEFENINGEN

- Demo 1: Som
- Demo 2: Min en max of array
- Demo 3: Retourneer array van randomwaarden
- A: Absolute waarde
- A: Afstand
- A: Aantrekkingskracht
- A: Palindroom
- A: From string to integer
- A: From hex to integer
- A: String match
- A: Sorteert array
- A: Mediaan
- A: Gemiddelde
- E: Waveforms
- X: De komeet
- X: Planetendans

### 2.1 Demo 1: Som

Schrijf een programma dat een som uitvoert van 2 ingegeven getallen. Deze som wordt weliswaar in een aparte functie uitgevoerd. Het doorgeven van argumenten en het retourneren van een waarde worden hierbij aangetoond.

## 2.2 Demo 2: Min en max of array

In deze opgave worden de kleinste en grootste waarde uit een array gehaald. Schrijf 2 functies waarbij de ene functie de kleinste waarde teruggeeft. De andere functie geeft de grootste waarde terug. Beide functies nemen als argument de array op.

## 2.3 Demo 3: Retourneer array van randomwaarden

In deze opgave schrijf je een functie die een array van 10 elementen aanmaakt en deze array ook retourneert. Print de array vervolgens af in een label gebruikmakend van een procedure.

## 2.4 A: Absolute waarde

C# beschikt over een functie die de absolute waarde van een getal kan bepalen. In deze opgave wordt er van je verwacht dat je zelf deze opgave als oefening maakt. Schrijf een functie die de absolute waarde van een getal uitrekt.

## 2.5 A: Afstand

Schrijf een functie die de afstand berekent tussen 2 punten (telkens x en y). De functie retourneert een waarde van het type double.

## 2.6 A: Aantrekkingskracht

Schrijf een functie die de aantrekkingskracht tussen 2 hemellichamen berekent. Gebruik hierbij de functie om afstanden te berekenen van voorgaande opgave. De functie die de aantrekkingskracht uitrekt neemt volgende argumenten op:

- x- en y positie van de hemellichamen,
- de massa's van de hemellichamen.

De functie retourneert een waarde die op het formulier uitgedrukt wordt in Newton.

## 2.7 A: Palindroom

Schrijf een functie die bepaalt of een woord een palindroom is of niet. Een palindroom is een woord dat in beide leesrichtingen (links naar rechts en rechts naar links) hetzelfde woord oplevert. Een aantal voorbeelden zijn:

- radar,
- ada,
- lepel,
- droomoord,
- madam,

- koortsmeetsysteemstrook,
- legermeetsysteemregel.

Zorg ervoor dat je ook palindromen kan detecteren die niet hoofdletter gevoelig zijn. De functie neemt de string aan als argument en retourneert een boolean.

**Hint:** Zet alle tekens om naar hoofdletters om de hoofdlettergevoeligheid kwijt te spelen.

## 2.8 A: From string to integer

In C# kan men de methode “int.Parse” gebruiken om een string om te zetten naar een integer. In deze opgave schrijf je zelf deze functie als oefening. Schrijf een functie die een string als argument opneemt en een integer teruggeeft. Hierbij ga je in de functie zoals hieronder te werk.

- Itereer over alle karakters van de string.
- Converteer elk teken naar een decimaal. Dit doe je door het karakter te casten naar een integer en te verminderen met 48. Zie ook de ascii-tabel voor de conversie.
- Vermenigvuldig het bestaande getal met 10 vooraleer het nieuwe bekomen cijfer op te tellen.

## 2.9 A: From hex to integer

Schrijf een functie die een hexadecimaal getal omzet naar een integer getal. Het hexadecimaal getal wordt ingegeven als een string. Ga karakter per karakter na om de juiste omzetting te bepalen. Ook hier kan je de ascii tabel gebruiken om een efficiënte omzetting te kunnen uitvoeren.

- Getallen van 0 t.e.m. 9: naar integer casten en 48 ervan afnemen.
- Letters van a t.e.m. f: naar integer casten en 97 ervan aftrekken. Tel nadien het getal 10 erbij op.

De algemene werkwijze verschilt niet veel van de voorgaande opgave.

## 2.10 A: String match

Schrijf een functie die een string in een andere string kan vinden. De functie neemt hierbij 2 strings als argumenten aan. De eerste string is een lange zin terwijl de 2<sup>de</sup> string een woord is. In de functie wordt het eerste voorkomen van het woord in de grote zin gezocht. De functie retourneert dan ook de positie (integer, index) van de zin van waarop het woord start. Hieronder worden enkele voorbeelden gegeven.

zin	woord	start index
Dit is een test	test	11
Is dat een grap?	een	7
Het is nu verlengd naar 8 seconden, omdat de track iets langer geworden is.	naar	19

Tabel 1: Voorbeelden van zinnen, woorden en de start positie. Merk op dat de array indices altijd bij 0 starten.

## 2.11 A: Sorteert array

Schrijf een functie die als argument een array van doubles opneemt. Deze functie sorteert de array en retourneert een array van doubles. Gebruik hierbij het algoritme van bubble sort. Bubble Sort is een eenvoudig algoritme dat toelaat om een rij waarden van klein naar groot te sorteren. Het algoritme werkt als volgt:

1. Start bij het begin van de rij.
2. Vergelijk de 1<sup>ste</sup> waarde met de 2<sup>de</sup>. Indien de 2<sup>de</sup> waarde kleiner is dan de 1<sup>ste</sup>, worden beide waarden gewisseld binnen de array.
3. Vergelijk nu de 2<sup>de</sup> met de 3<sup>de</sup> waarde en voer dezelfde wisseloperatie uit indien nodig.
4. Ga zo door totdat alle elementen van de array afgegaan zijn.
5. Start opnieuw bij het begin (punt 2) en herhaal dit totdat alle elementen zijn afgegaan.

Schrijf ook een functie die toelaat om een array van doubles met randomgetallen te vullen. Deze functie retourneert ook een array die gebruikt wordt als argument voor de sorteerfunctie. Als array mag je een lengte nemen tussen 10 en 25 elementen.

## 2.12 A: Mediaan

Herneem voorgaande opgave. De oplossing mag in dezelfde solution geschreven worden. Schrijf een nieuwe functie die toelaat om van de gesorteerde array de mediaan te bepalen. Zoek het middelste element en retourneer dit. indien de array even is van lengte, neem je de 2 middelste elementen en bereken je daarvan het gemiddelde. In dat geval retourneer je het gemiddelde van de 2 middelste elementen.

## 2.13 A: Gemiddelde

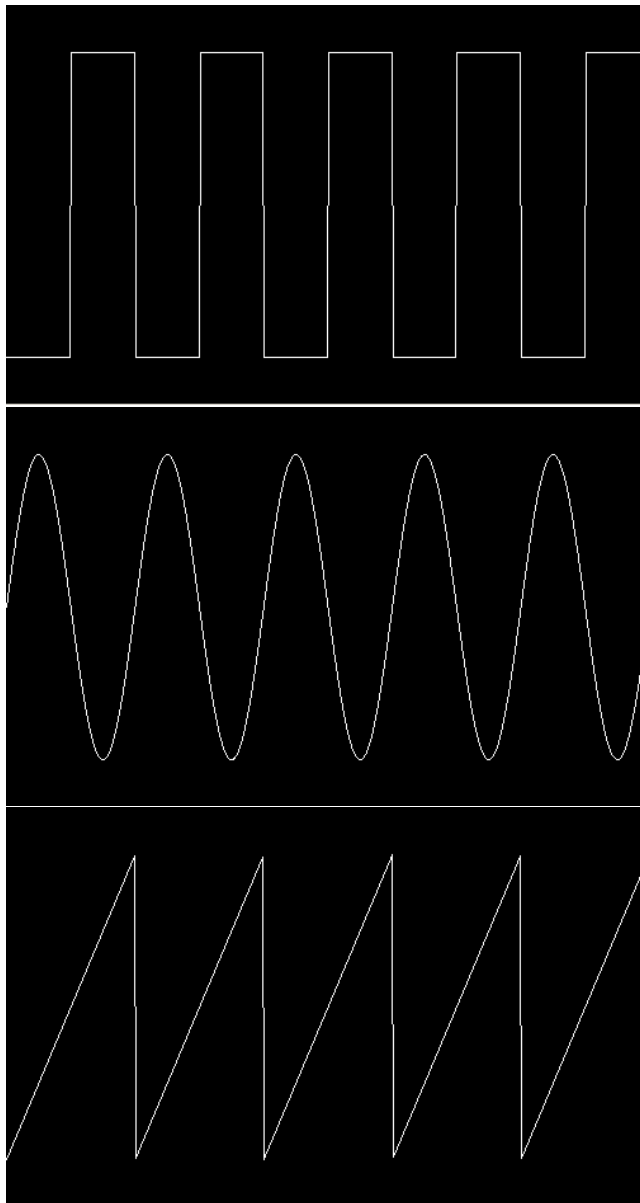
Herneem voorgaande opgave. De oplossing mag in dezelfde solution geschreven worden. Schrijf een functie die het gemiddelde van de array bepaald en teruggeeft. Print nadien de waarde af in een label.

## 2.14 E: Waveforms

Schrijf een programma waarin je 4 functies voorziet. De eerste 3 functies berekenen een geluidsgolf terwijl de 4<sup>de</sup> functie een geluidsgolf tekent. De te tekenen geluidsgolven zijn:

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

Deze worden weergegeven in afbeelding 1.



**Figuur 1:** Voorbeelden van een blokgolf, zaagtand en sinusgolf.

Elk van deze functies retourneert een array met daarin de samples. Elk 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.

## 2.15 X: De komeet - opgelet: OOP - voor de echte durvers

In ons zonnestelsel bevinden zich een zeer groot aantal hemellichamen, waaronder onze vertrouwde zon en enkele kometen. Afgezien van een aantal niet gemeenschappelijke eigenschappen worden hemellichamen gekenmerkt door de massa, de grootte (diameter), de bewegingssnelheid, de kleur en de positie (2D). Maak een klasse aan dat deze eigenschappen van de hemellichaam bevat. In ons zonnestelsel worden alleen de zon en één komeet getekend. Teken beide hemellichamen door deze beide de gepaste eigenschappen te geven.

Natuurlijk blijven hemellichamen niet stationair binnen het zonnestelsel evolueren. Deze bewegen. De beweging van de hemellichamen wordt voornamelijk bepaald door de huidige snelheid van het hemellichaam en de zwaartekracht dat deze ondervindt t.g.v. andere hemellichamen. De grootte van de zwaartekracht t.g.v. een ander hemellichaam wordt beschreven als:

$$F = G \cdot \frac{m_1 \cdot m_2}{r^2} \quad (1)$$

waarbij  $F$  de grootte van de zwaartekracht is,  $m_1$  de massa van hemellichaam 1 is,  $m_2$  de massa van hemellichaam 2 en  $r$  de afstand tussen beide hemellichamen. Ga dus na hoe de afstand tussen beide hemellichamen berekend kan worden (maak hiervoor een functie aan).

In deze opgave mag de zon statisch blijven, dus enkel de positie van de komeet verandert. De nieuwe positie en snelheid van de komeet worden als volgt bepaald:

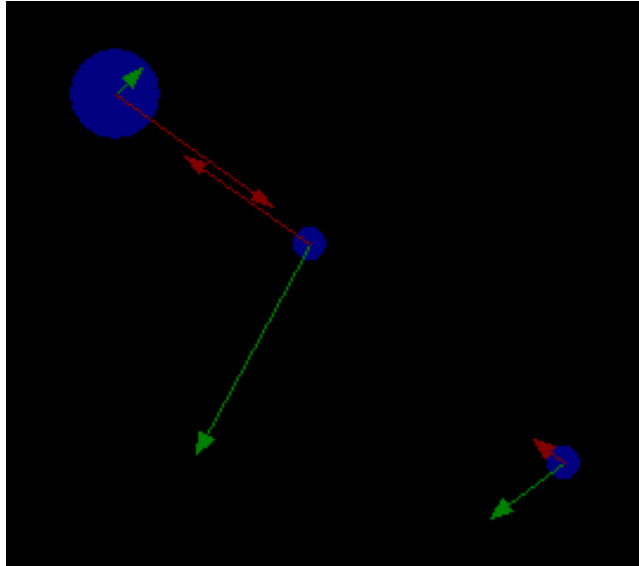
$$p_{komeet} = p_{komeet} + v_{komeet} \cdot t + \frac{at^2}{2} \quad (2)$$

$$v_{komeet} = v_{komeet} + at \quad (3)$$

De zwaartekracht, snelheid en positie worden uitgedrukt in vectoren (dus X en Y). Het verband tussen het vectorieel verband en de numerieke waarden in formules 1, 2 en 3 wordt gegeven door de hoek tussen de posities van de zon en de komeet.

Gebruik een timer om het effect van de gravitatie op de komeet weer te geven. Teken ook de situatie.

**Hint:**  $F = m \cdot a$



**Figuur 2:** Voorbeeld van 3 hemellichamen. Hier worden ook de snelheidsvector (groen), samen met de zwaartekracht (rood) dat ondervonden wordt door elk lichaam getekend. De pijlen hoeven niet getekend te worden.

## 2.16 X: Planetendans - opgelet: OOP - voor de echte durvers

Deze opgave is een uitbreiding van de voorgaande opgave. Nu worden meer dan 2 hemellichamen getekend. Bovendien bewegen alle hemellichamen, wat een realistischer resultaat als gevolg heeft. De gravitatie dat een hemellichaam ondervindt t.g.v. de andere hemellichamen worden nu gegeven door de vectoriële som van de afzonderlijke gravitaties. Sla alle hemellichamen op in een array, en doorloop de array om alle hemellichamen te behandelen. De opgave is voorts identiek aan de voorgaande opdracht.