

Oefening 1

- Maak een array die groot genoeg is om 20 gehele getallen tussen 1 en 10 te bevatten
- Laat de gebruiker 20 keer een getal kiezen uit het interval [1,10] (controleer dit!) en vul de array op met deze getallen.

```
Geef het 1e getal uit het interval [1-10]: 0
Geef het 1e getal uit het interval [1-10]: -4
Geef het 1e getal uit het interval [1-10]: 20
Geef het 1e getal uit het interval [1-10]: 7
Geef het 2e getal uit het interval [1-10]: 4
Geef het 3e getal uit het interval [1-10]: 8
Geef het 4e getal uit het interval [1-10]: 2
Geef het 5e getal uit het interval [1-10]: 7
Geef het 6e getal uit het interval [1-10]: 5
Geef het 7e getal uit het interval [1-10]: 3
Geef het 8e getal uit het interval [1-10]: 6
Geef het 9e getal uit het interval [1-10]: 1
Geef het 10e getal uit het interval [1-10]: 10
Geef het 11e getal uit het interval [1-10]: 8
Geef het 12e getal uit het interval [1-10]: 10
Geef het 13e getal uit het interval [1-10]: 3
Geef het 14e getal uit het interval [1-10]: 2
Geef het 15e getal uit het interval [1-10]: 7
Geef het 16e getal uit het interval [1-10]: 9
Geef het 17e getal uit het interval [1-10]: 7
Geef het 18e getal uit het interval [1-10]: 3
Geef het 19e getal uit het interval [1-10]: 3
Geef het 20e getal uit het interval [1-10]: 2
```

- Hou in een tweede array (bepaal zelf de geschikte grootte!) bij hoeveel keer elk getal uit het interval voorkomt.
- Toon de getallen uit de array met ingevoerde getallen op 1 rij, gescheiden door een spatie (zie ook voorbeelduitvoer).
- Toon een overzichtstabel met per getal uit het interval [1,10] het aantal keer dat het in de array met ingevoerde getallen voorkomt (zie ook voorbeelduitvoer).

De array bevat volgende getallen:

7 4 8 2 7 5 3 6 1 10 8 10 3 2 7 9 7 3 3 2

Dit levert volgende telling op:

GETAL	AANTAL KEER
1	1
2	3
3	4
4	1
5	1
6	1
7	4
8	2
9	1
10	2

Oefening 2

Gegeven:

een array a die een willekeurig aantal gehele getallen bevat met een willekeurige waarde
Vb. `int[] a = {4,7,15,3,9,22,36,24,28,14,19,27,30,31,2,9,29,30,16,19};`

Gevraagd:

Bepaal voor deze array de langste reeks opeenvolgende getallen in stijgende lijn en toon zowel het aantal getallen in deze reeks als de reeks zelf (de getallen, gescheiden door een spatie)

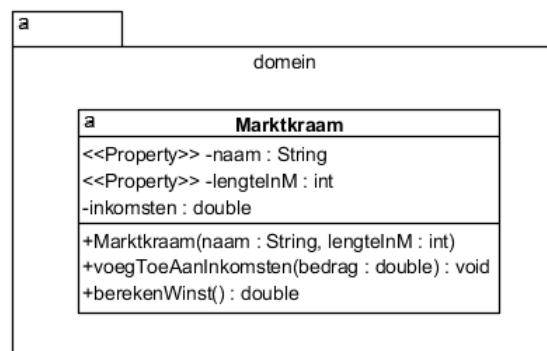
Opmerking: het programma moet (vanzelfsprekend) ook met een andere array dan die uit bovenstaand voorbeeld blijven werken

Voorbeeldoutput:

Het meest aantal opeenvolgende stijgende getallen is 5 en bestaat uit de reeks 14 19 27 30 31

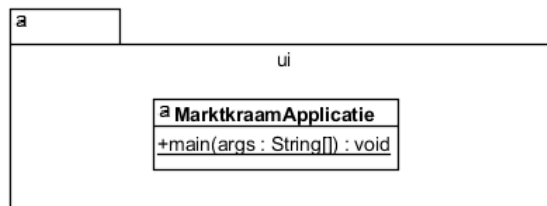
Oefening 3

Maak een domeinklasse **Marktkraam** met volgende functionaliteiten:



- **3 attributen:** de naam van de uitbater, de lengte in meter (geheel getal) en de inkomsten (kommagetal)
- **Constructor:** stelt de naam en de lengte in
- **Setters voor naam en lengteInM:** naam moet ingevuld zijn en lengte moet in het interval [2,10] liggen. Wanneer de parameterwaarde niet voldoet, wordt als naam “anoniem” en als lengte 10 meter gekozen.
- Methode **voegToeAanInkomsten:** controleert of de meegegeven parameter een positief bedrag bevat. Zo ja, wordt het bedrag opgeteld bij de inkomsten
- Methode **berekenWinst** berekent eerst de kostprijs van de kraam: 3 euro per lopende meter voor kramen tot en met 5 meter lang of 2,5 euro indien de kraam meer dan 5 meter lang is. Daarna wordt de winst berekend: hiervoor wordt van de inkomsten deze kostprijs afgetrokken.

In de applicatieklasse **MarktkraamApp** wordt een rommelmarkt gesimuleerd (zie ook voorbeelduitvoer):



- Eerst wordt het aantal kramen opgevraagd. Dit aantal moet strikt positief zijn, anders wordt het opnieuw gevraagd.

```

Geef het aantal kramen op de rommelmarkt: -1
Geef het aantal kramen op de rommelmarkt: 3
    
```

- Aan de hand van dit aantal wordt een array van Marktkraam-object gemaakt.
- Per Marktkraam wordt nu gevraagd wie de uitbater is van de kraam en hoeveel lopende meter zijn kraam inneemt. Met deze gegevens wordt een object gemaakt waar in een van de arrayelementen naar wordt gerefereerd.

```

Geef de naam van de eigenaar van kraam 1: Jan
Geef de afmeting in meter van kraam 1: 5
Geef de naam van de eigenaar van kraam 2: Piet
Geef de afmeting in meter van kraam 2: 8
Geef de naam van de eigenaar van kraam 3: Els
Geef de afmeting in meter van kraam 3: 3
    
```

- Daarna mogen er inkopen gedaan worden. De gebruiker krijgt telkens een genummerde lijst van de aanwezige kramen en de mogelijkheid om te stoppen (keuze 0). Hij voert zijn keuze in en als deze naar een geldig kraamnummer verwijst, krijgt hij ook nog de vraag om de prijs van het gekochte item op te geven. Deze prijs wordt dan aan de inkomsten van die bepaalde kraam toegevoegd.

```

Aan welke kraam wil je iets kopen?
0. Stoppen
1. Kraam van Jan
2. Kraam van Piet
3. Kraam van Els
Geef je keuze: 1
Geef de prijs in euro van het gekochte item: 5
Aan welke kraam wil je iets kopen?
0. Stoppen
1. Kraam van Jan
2. Kraam van Piet
3. Kraam van Els
Geef je keuze: 3
Geef de prijs in euro van het gekochte item: 1
Aan welke kraam wil je iets kopen?
0. Stoppen
1. Kraam van Jan
2. Kraam van Piet
3. Kraam van Els
Geef je keuze: 2
Geef de prijs in euro van het gekochte item: 0,5
    
```

```

Aan welke kraam wil je iets kopen?
0. Stoppen
1. Kraam van Jan
2. Kraam van Piet
3. Kraam van Els
Geef je keuze: 0

```

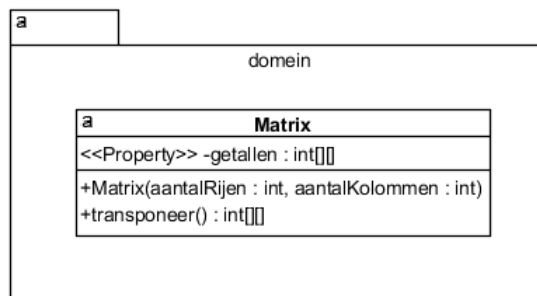
- Wanneer de gebruiker optie 0 gekozen heeft, eindigt het programma met nog een overzicht van alle kramen en hun gemaakte winst in tabelvorm.

Overzicht van de winst per kraam

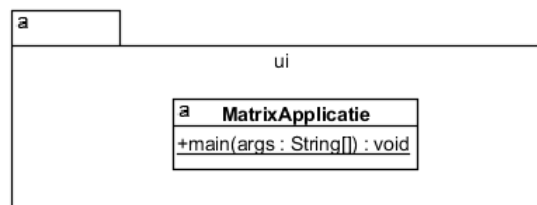
1	Jan	17,00 euro
2	Piet	-16,50 euro
3	Els	29,20 euro

Oefening 4

Maak een domeinklasse **Matrix** die een tweedimensionale array van gehele getallen kan bijhouden. Bij constructie wordt het aantal rijen en kolommen van deze matrix gecontroleerd (moet minstens 1 zijn, defaultwaarde 1) en ingesteld. Er is een **getter en setter** voor het attribuut en er is een methode **transposeer** die een nieuwe 2-dim array teruggeeft waarin de rijen en kolommen van de oorspronkelijke matrix omgewisseld zijn.



Maak een applicatieklasse **MatrixApplicatie** waarin de gebruiker eerst het aantal rijen en kolommen van de matrix opgeeft. Aan de hand hiervan wordt er een Matrix-object gemaakt. Daarna worden de elementen van de Matrix rij per rij en kolom per kolom opgevraagd. Als alle elementen ingelezen zijn, worden deze toegekend aan het Matrix-object. Daarna wordt de matrix getransponeerd en tenslotte wordt de getransponeerde matrix getoond.



Voorbeelduitvoer:

```
Geef het aantal rijen van de matrix: 4
Geef het aantal kolommen van de matrix: 2
Geef het element op rij 1 en kolom 1: -1
Geef het element op rij 1 en kolom 2: 3
Geef het element op rij 2 en kolom 1: 5
Geef het element op rij 2 en kolom 2: -2
Geef het element op rij 3 en kolom 1: 7
Geef het element op rij 3 en kolom 2: 0
Geef het element op rij 4 en kolom 1: -5
Geef het element op rij 4 en kolom 2: 4
De getransponeerde matrix:
  -1    5    7   -5
   3   -2    0    4
```