

## Assignment 42 – Tic Tac Toe - Basic

### Het doel

Maak een Java console applicatie waarmee het bekende spel Tic Tac Toe (boter-kaas-en-eieren) mee gespeeld kan worden. De applicatie dient procedureel worden geprogrammeerd.

Het doel van de opdracht is het toepassen van de geleerde onderwerpen die tijdens de training zijn besproken en begrijpen hoe een “eenvoudige” applicatie als Tic Tac Toe werkt.

### Woord vooraf

Lees eerst de opdracht goed door en probeer eerst uit te denken hoe je het programma gaat opzetten. Maak eventueel een “schets” op papier, een flow chart kan helpen!

Het programma dient uitsluitend 1 class te hebben met de naam TicTacToeBasic waarin de main methode is opgenomen. Geen extra classes zijn nodig, maar uiteraard kun je mooie methodes bedenken die de code compacter en efficiënter maken.

Voorzie commentaar in je code om voor jezelf en anderen duidelijk te maken wat je op bepaalde momenten doet. Voor de docenten helpt dit bij het nakijken.

Heel veel succes!

## Introductie

Tic Tac Toe is een eenvoudig spel dat meestal met papier en potlood wordt gespeeld. Eerst maak je een eenvoudig 3 x 3 raster (spelbord) op het papier. Vervolgens wisselen twee spelers beurten af door kruisjes en rondjes te markeren in lege vakjes op het rooster. De eerste speler die drie van zijn of haar markeringen in een horizontale, verticale of diagonale rij maakt, wint. Als alle velden in het rooster zijn ingevuld voordat een speler drie op een rij heeft gemarkeerd, is het spel gelijkspel.

## Regels en Instructies

1. De computer speelt tegen een menselijke speler. De speler begint en heeft het symbool XX. De computer heeft symbool OO.
2. Het programma begint met het tonen van een welkomstbericht.
3. Het spelbord is een 3 x 3 raster met de kolommen A, B en C en rijen 1, 2 en 3. Dit zijn de posities waar gespeeld kan worden.
4. Daarna wordt de menselijke speler gevraagd om de eerste zet.
5. Te verduidelijking is hieronder de opening van het spel afgebeeld:

```
Welcome to Tic-Tac-Toe.

A1 | B1 | C1
---
A2 | B2 | C2
---
A3 | B3 | C3

Please enter your first move:
```

Om een X in het vak linksboven te plaatsen, voert de menselijke speler dus de tekst A1 in wanneer het programma de menselijke speler om een zet vraagt. Nadat de menselijke speler een zet heeft ingevoerd, moet het programma de huidige status van het bord op de console weergeven.

6. Gebruik XX om de gespeelde zetten van de menselijke speler te markeren en OO om de zetten van de computer te markeren. Gebruik verticale balktekens (te vinden op het toetsenbord boven het backslash-teken) en koppeltekens om het bord in een eenvoudig raster te tekenen. Als de menselijke speler bijvoorbeeld A1 heeft ingevoerd als zijn eerste zet, geeft het programma het volgende weer:

```
Please enter your first move: A1

XX |  | 
---
  |  | 
---
  |  |
```

7. Na de zet van de menselijke speler, bepaalt het programma (de computer) zijn zet, kondigt het aan de menselijke speler aan, toont een bijgewerkt bord en vraagt vervolgens naar de zet van de menselijke speler. Je zou dit bijvoorbeeld kunnen zien:

```
I will play at A2

XX |  | 
---
00 |  | 
---
  |  |

Please enter your next move:
```

8. Het spel gaat door totdat een speler drie op een rij heeft gescoord (horizontaal, verticaal of diagonaal) of alle posities zijn gevuld zonder winnaar. Je programma moet kunnen bepalen of een van de spelers drie keer op rij heeft gescoord en het spel heeft gewonnen. (Dit is het moeilijkste deel van deze programmeeruitdaging.)
9. Wanneer het spel is afgelopen, geeft het programma een bericht weer met het resultaat van het spel: "You beat me!" Als de menselijke speler wint, "I beat you!" Als de computerspeler wint, of "It's a draw!" als het spel eindigt in een gelijkspel.
10. De menselijke en computerspelers kunnen alleen spelen op de posities die nog niet door 1 van beide spelers is gezet.
11. Het programma eindigt wanneer het spel door een van beide spelers wordt gewonnen of het een gelijkspel is. Als je opnieuw wilt spelen, moet je het programma opnieuw uitvoeren.

Merk op dat je vrij bent om elke methode te gebruiken die je wenst te bepalen hoe de computer zijn bewegingen moet maken. Ik stel voor dat je de computer altijd in het eerste lege vakje laat spelen. Dit is duidelijk niet de beste manier om Tic Tac Toe te spelen, en je zult geen moeite hebben om de computer elke keer dat je speelt te verslaan. Maar door deze eenvoudige speelstrategie te kiezen, kun je je concentreren op andere aspecten van de programmering, zoals hoe je het raster intern kunt weergeven en hoe je kunt bepalen wanneer een speler het spel heeft gewonnen of wanneer het spel is geëindigd in een gelijkspel.

Hier is een voorbeeld van console output voor een compleet gespeeld spel:

```
Welcome to Tic-Tac-Toe.

  A1 | B1 | C1
  ---
  A2 | B2 | C2
  ---
  A3 | B3 | C3

Please enter your first move: A1

  XX |   |
  ---
   |   |
  ---
   |   |

I will play at A2

  XX |   |
  ---
  00 |   |
  ---
   |   |

Please enter your next move: B2

  XX |   |
  ---
  00 | XX |
  ---
   |   |

I will play at A3

  XX |   |
  ---
  00 | XX |
  ---
  00 |   |

Please enter your next move: C3

  XX |   |
  ---
  00 | XX |
  ---
  00 |   | XX

You beat me!

Process finished with exit code 0
```

## Hints

Hier zijn enkele tips om je op weg te helpen:

- De beste manier om het raster weer te geven is met een array, maar voor het gemak gebruik voorlopig negen rastervariabelen, genaamd A1, A2, A3, B1, B2, B3, C1, C2 en C3.
- Als je het type int gebruikt voor de rastervariabelen, kun je vervolgens 1 gebruiken om aan te geven dat een positie XX bevat en 2 om aan te geven dat een positie OO bevat.
- Acht mogelijke rijen kunnen leiden tot een overwinning. Een speler wint als een van de volgende combinaties van posities ofwel 1 (voor XX) of 2 (voor OO) zijn:

```
A1 - A2 - A3
B1 - B2 - B3
C1 - C2 - C3
```

```
A1 - B1 - C1
A2 - B2 - C2
A3 - B3 - C3
```

```
A1 - B2 - C3
A3 - B2 - C1
```

- Er zijn twee manieren om te bepalen of het gelijkspel is. De eerste is om alle negen vakjes te controleren: als geen van de vakjes leeg is en geen van de spelers heeft gewonnen, is het gelijkspel. De tweede is het tellen van de zetten: als er negen zetten zijn gedaan en geen van beide spelers heeft gewonnen, is het gelijkspel.