

## Opdracht 1 - Gegevensopslag

a) Maak een `class Persoon` met de velden:

- naam (`string`),
- woonplaats (`string`) en
- leeftijd (`int`).

Maak een methode:

`Persoon` `LeesPersoon()`

Deze methode vraagt om de gegevens van een persoon, leest deze in en retourneert een `Persoon`-object.

Maak ook een methode:

`void` `ToonPersoon(Persoon p)`

Deze methode toont de gegevens van een persoon op het scherm.

→ Test beide methoden door ze vanuit de `Start`-methode aan te roepen.

b) We willen de ingelezen persoon naar een bestand kunnen wegschrijven zodat we een volgende keer niet nogmaals deze gegevens moeten invoeren.

Maak daarvoor een methode:

`void` `SchrijfPersoon(Persoon p, string bestandsNaam)`

Deze methode schrijft de persoon naar het opgegeven bestand (elk veld komt op een aparte regel).

→ Test de methode in de `Start`-methode.

→ Kijk of het bestand is aangemaakt en of het de verwachte inhoud bevat.

c) Maak nu een methode:

`Persoon` `LeesPersoon(string bestandsNaam)`

Deze methode leest een persoon uit het opgegeven bestand en retourneert deze.

→ Test de methode in de `Start`-methode.

d) Gebruik de gemaakte methoden om er voor te zorgen dat de `Start`-methode het volgende doet:

- De gebruiker wordt gevraagd om zijn/haar naam in te voeren.
- [Nieuwe gebruiker]: Wanneer er geen bestand voor deze gebruiker bestaat (<naam>.txt'), dan wordt de gebruiker gevraagd zijn/haar gegevens in te voeren en worden deze gegevens opgeslagen in een bestand (met bestandsnaam '<naam>.txt').
- [Bestaande gebruiker]: Wanneer er een bestand bestaat met de naam van de gebruiker (bv 'Kees.txt'), dan wordt de melding 'Wat leuk om je weer te zien <naam>' getoond en daaronder de gegevens die in het bestand staan.

```
Wat is uw naam? Nelleke
Welkom nieuweling!
Hoe heet u? Nelleke
Waar woont u? Voorschoten
Hoe oud bent u? 22
Uw gegevens zijn opgeslagen
```

```
Wat is uw naam? Nelleke
Wat leuk om u hier weer te zien!
Wij hebben de volgende informatie over u:
Naam      : Nelleke
Woonplaats: Voorschoten
Leeftijd  : 22
```

## Opdracht 2 – Galgje

De laatste versie van Galgje werkte met een hardcoded woordenlijst. Het is natuurlijk leuker om een woord te kiezen uit een lange woordenlijst (uit een file). Je moet dan wel weer eerst alle woorden inlezen en dan pas random bepalen met welk woord het spel wordt gespeeld.

a) Pas het programma aan, zodat de woorden uit een tekstbestand worden gelezen. Gebruik het bestand "woorden.txt" dat je op Moodle kunt vinden. Zorg uiteraard dat je programma blijft werken.

**Let op:** Het bestand op Moodle bevat ook hele korte woorden. Controleer dus altijd of het gekozen woord minstens drie letters lang is. Zo niet, dan wordt een ander woord gekozen.

## Opdracht 3 – Woordzoeker

Bij deze opdracht voert de gebruiker een woord in dat vervolgens in een gegevensbestand wordt gezocht. Alle regels die het opgegeven woord bevatten worden op het scherm getoond. Een bestand met tweets van president Donald Trump vind je op Moodle (maar je mag ook een ander tekstbestand gebruiken).

- a) Maak een methode met signatuur

```
bool ZitWoordInRegel(string regel, string woord)
```

Om te bepalen of een woord in een regel voorkomt. Let er op dat je 'case insensitive' zoekt.

- b) Maak een methode:

```
int ZoekWoordInBestand(string bestandsnaam, string woord)
```

Deze methode leest een bestand regel voor regel.

Wanneer een regel het opgegeven woord bevat (gebruik methode ZitWoordInRegel), dan wordt de regel op het scherm afgedrukt.

De methode ZoekWoordInBestand retourneert het aantal regels dat het woord bevat.

De Start-methode toont dit aantal op het scherm.

```
file:///C:/Users/Gerwin van Dijken/Documents/Visual Studio 2015/Projects/PraktijkProgrammeren...
Enter a word (to search): christmas

Twitter for iPhone,On behalf of @FLOTUS Melania and the entire Trump family I want to wish you all a very MERRY CHRISTMAS! May this Christmas Season bring peace to your hearts warmth to your homes cheer to your spirits and JOY TO THE WORLD! #NCTL2018?? https://t.co/XNMJQ5JDSU,11-28-2018 23:32:11,28019,117362,false,1067924142316494848
Twitter for iPhone,RT @FLOTUS: The @WhiteHouse is getting ready for the Christmas se
```

- c) Om aan te geven waar het zoekwoord zich bevindt in een regel, toon het zoekwoord in een andere kleur.

Maak een methode:

```
void ToonWoordInRegel(string regel, string woord)
```

Deze methode toont de opgegeven regel op het scherm, waarbij het opgegeven woord in het rood wordt weergegeven (Zie screenshot hieronder).

Gebruik regel.IndexOf(woord) om de start-positie van het woord in de regel te bepalen.

Gebruik regel.Substring(start) of regel.Substring(start, aantal) om een stuk uit de regel te knippen.<sup>1</sup>

Het volstaat om alleen het eerst gevonden woord in het rood te tonen.

→ Wijzig methode ZoekWoordInBestand zodat de regels via methode ToonWoordInRegel worden getoond

→ [optioneel] Toon alle voorkomens van het woord in het rood (zie laatste screenshot).

```
file:///C:/Users/Gerwin van Dijken/Documents/Visual Studio 2015/Projects/PraktijkP...
Twitter for iPhone,RT @WhiteHouse: President Trump and the First Lady Participate in the White House Christmas Tree Delivery https://t.co/O6Nr8k1mnu,11-19-2018 20:56:08,11977,0,true,1064623382359085060
Number of lines containing the word: 3
```

```
file:///C:/Users/Gerwin van Dijken/Documents/Visual Studio 20...
Enter a word (to search): christmas

Twitter for iPhone,On behalf of @FLOTUS Melania and the entire Trump family I want to wish you all a very MERRY CHRISTMAS! May this Christmas Season bring peace to your hearts warmth
Twitter for iPhone,RT @FLOTUS: The @WhiteHouse is getting ready for the Christmas se
```

<sup>1</sup> Zie Yellow-book § 4.11 The power of strings and chars.

## Opdracht 4 – Candy Crush

We gaan de CandyCrush opdracht van de 2<sup>e</sup> week zodanig aanpassen, dat we het speelveld kunnen opslaan in een bestand en dat we het spel ook kunnen starten op basis van een opgeslagen speelveld (door het te lezen).

a) Maak een methode:

```
void SchrijfSpeelveld(RegularCandies[,] speelveld, string bestandsnaam)
```

Deze methode slaat het speelveld op in een bestand. Gebruik hiervoor de `int`-waarden van de enumeratie `RegularCandies`. Sla de waarden rij-voor-rij op zodat wanneer je het bestand opent in een editor het speelveld ook herkenbaar is. Een mogelijk formaat zie je in onderstaand figuur.

b) Maak een methode:

```
RegularCandies[,] LeesSpeelveld(string bestandsnaam)
```

Deze methode leest het speelveld uit een bestand (met het opgegeven bestandsnaam) en retourneert dit speelveld.

**Tip:** Om meerdere getallen van één regel in te lezen, lees je eerst de hele regel in als string. Deze knip je in een array met string-waarden (m.b.v. `Split`). De strings in dat array kan je dan één voor één parsen naar een `int`-waarde.

```
→ string[] getalStrings = regel.Split(' ');
```

c) Wijzig nu de `Start`-methode zodat het speelveld uit een bestand wordt gelezen (als het bestand bestaat) of gevuld wordt met random candies (als het bestand niet bestaat). Wanneer een nieuw speelveld is gegenereerd, wordt dit weggeschreven naar een bestand.

d) Als het bestand niet correct is (er staan bijvoorbeeld niet genoeg regels in, of een regel heeft niet genoeg getallen), dan mag het programma niet crashen. Vang in de `Start`-methode de exception op, en genereer vervolgens een nieuw speelveld met random candies (en sla deze op).

e) Je kunt nu het opgeslagen speelveld wijzigen (in het bestand, via bv Notepad) om vervolgens de methoden `ScoreRijAanwezig` en `ScoreKolomAanwezig` met specifieke waarden te testen. Er zijn bepaalde 'grenswaarden' die (hoogstwaarschijnlijk) niet in een random gegenereerd speelveld zitten, maar die je nu zelf kunt vastleggen in een bestand. Wat zijn volgens jou goede 'grenswaarden' om te testen?

→ Test deze grenswaarden.

```
playingfield.txt ...
File Edit Format View Help
9
9
3 6 2 3 2 3 3 5 6
3 5 2 3 4 4 5 2 6
6 6 3 4 3 2 5 4 6
2 1 1 3 2 3 4 3 1
5 6 6 4 5 1 3 2 1
1 2 6 4 6 3 5 1 5
3 6 3 6 5 4 2 2 4
5 1 3 5 6 2 3 1 2
5 2 2 2 2 2 2 4 3
```

Mogelijk formaat voor een CandyCrush tekstbestand  
(met op de 1e rij het aantal rijen, op de 2e rij het  
aantal kolommen, en vervolgens <aantal rijen> rijen  
met elk <aantal kolommen> kolommen)