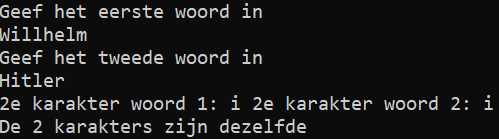
Oefeningen Strings

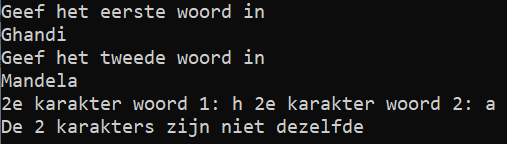
## Twee woorden (\*)

Vraag de gebruiker om 2 woorden in te geven. Wij gaan na of het karakter op de 2e plaats, hetzelfde karakter is.

Als output wordt het 2e karakter van ieder woord weergegeven alsook of ze dezelfde zijn of niet.

Bedenk ook het scenario als je een woord van 1 letter meegeeft.





## Aantal hetzelfde woord, aantal verschillend woord (\*)

Er kan een oneindig aantal keer 2 woorden in gegeven worden na elkaar. Eindigen van de invoer wordt gedaan door een “?”.

Als resultaat wordt er gegeven hoeveel keer hetzelfde woord na elkaar werd ingeven en hoeveel keer een verschillend woord.

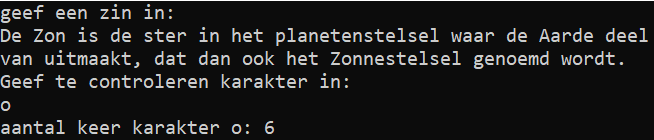


## Aantal keer dat een teken voorkomt in een zin (\*)

We lezen een zin in en we geven een karakter in waarvan we willen nagaan hoeveel keer het karakter in de zin voorkomt.

De output zal het aantal keer dat het ingegeven karakter voorkomt weergeven.

Gebruik een for-lus om ieder karakter van de ingegeven zin te overlopen.



## Keuzes op een zin (\*)

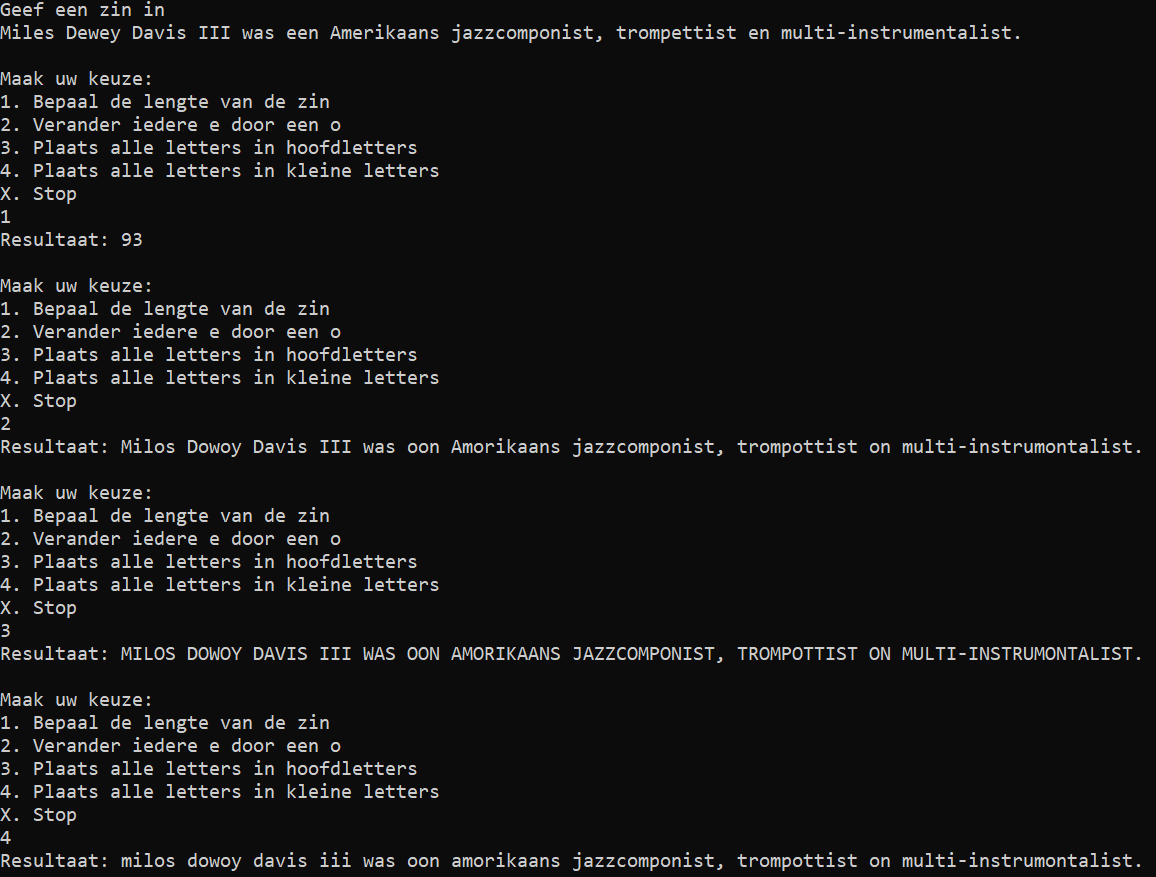
We maken een programma dat verschillende keuze kan maken op een zin die ingegeven wordt.

1. Bepaal de lengte van de zin
2. Verander iedere e door een o
3. Plaats alle letters in hoofdletters
4. Plaats alle letters in kleine letters
5. Vervang alleen de eerst keer dat de letter o voorkomt door Jantje

Afsluiten doe je door een “X” in te geven.

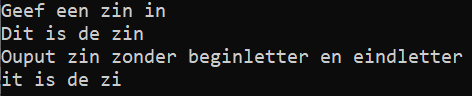
Zorg dat er altijd verder gewerkt wordt, met de aangepaste zin.

Gebruik voor de oefening minimaal één loop en minimaal een switch instructie.



## Verwijder 1ste en laatste karakter (\*)

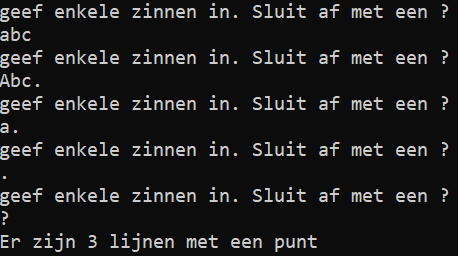
Geef een voorbeeldzin in. Verwijder de 1e en de laatste letter van de voorbeeldzin. Zorg ervoor dat gelijk welke lengte van woorden kan ingegeven worden.



## Aantal zinnen met een punt (\*)

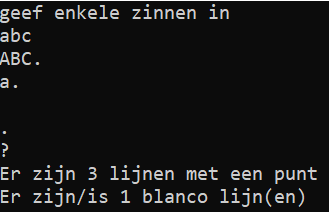
Schrijf een programma dat de regels die ingegeven worden zal weergeven. De invoer wordt beëindigd door het ingeven van een vraagteken. Tel hoeveel regels eindigen op een punt.

Gebruik hiervoor minimaal één loop.



## Aantal lijnen met een punt of blankolijn (\*)

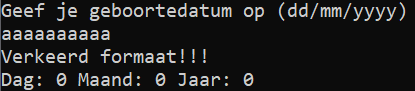
Breidt vorige oefening uit, zodat ook een blanco-lijn kan ingegeven worden. Het aantal blanco-lijnen komt ook in de output voor. Ga hierbij na hoe een blancolijn er gedetecteerd worden.

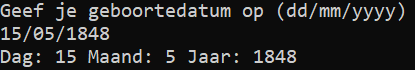


## Geboortedatum (\*)

Schrijf een programma dat aan de gebruiker zijn geboortedatum ingeeft in de vorm van dd/mm/yyyy als een String. Het ‘/’ teken moet de gebruiker dus ook opgeven. Het programma zal uit deze String de dag, maand en jaar extraheren. Hiervoor maken we gebruik van de Substring() methode. De data van de dag, maand en jaar moet wel als een integer worden bewaard.

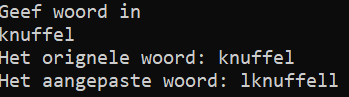
Bij het omzetten van string naar int, adhv de int.Parse method, gebruik een Try…Catch





## BijPlakken (\*)

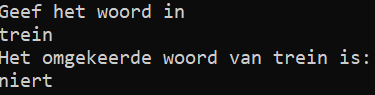
Schrijf een programma dat aan de gebruiker een woord opvraagt. Het programma zal de laatste letter van dat woord nemen en het zowel vóór het woord toevoegen als na het woord.



## StringOmgekeerd (\*)

Schrijf een programma dat aan de gebruiker een zin opvraagt. Het programma zal de zin in de omgekeerde volgorde terug op het scherm plaatsen.

Je mag deze karakter per karakter onmiddellijk uitlezen.

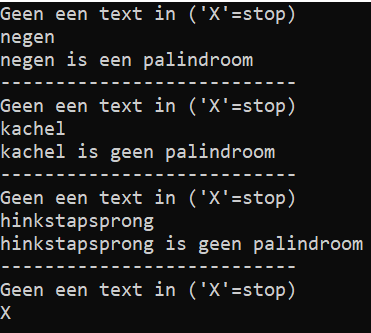


## Palindroom (\*\*)

Maak een programma die in staat is om te onderzoeken of de ingevoerde tekst een palindroom is. Een palindroom is bijvoorbeeld kok, lepel, parterretrap.

Los dit op zonder het woord om te draaien.

Vb. kok, pap, lepel, droomoord, legerregel, level, negen, lol, meetsysteem, legovogel, parterretrap



## Woord raden (\*\*\*)

Maak een programma dat een ingegeven woord raadt via de ingave van een letter.

Zorg ervoor dat het programma na het raden van een woord een nieuw woord vraagt. Het programma stopt wanneer men voor een nieuw woord de letter ”X” intikt.

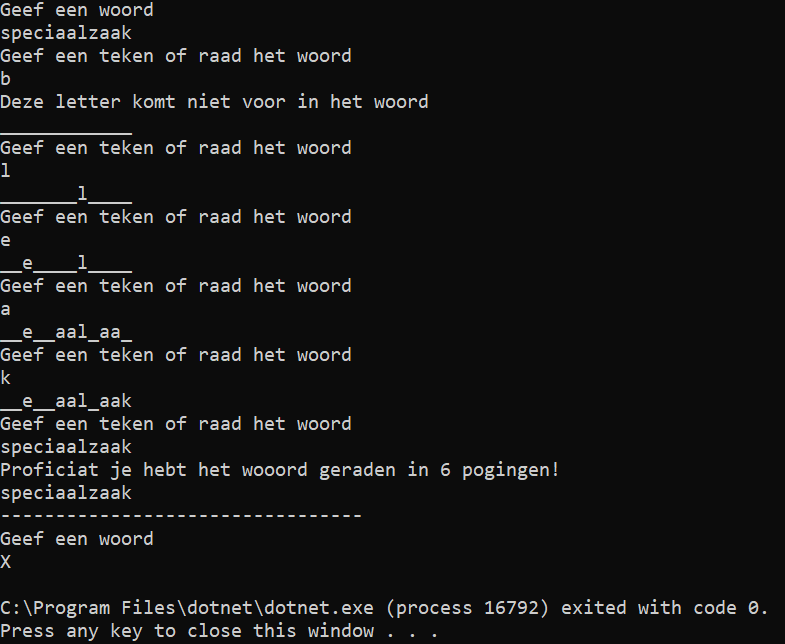
Het raden van een woord gebeurt aan de hand van de ingave van een letter of het geraden woord.

Het te raden woord wordt voorgesteld door een aantal “\_”-tekens. We tonen evenveel “\_”-tekens als er letters zijn in het te raden woord.

Komt het ingegeven teken voor in het woord, dan worden de “\_”-tekens vervangen door het ingegeven teken en men toont het woord met enkel de geraden tekens ingevuld.

Komt het ingegeven teken niet voor, dan moet men opnieuw een teken raden.

Is het woord geraden dan toont men op het scherm het aantal pogingen dat men nodig had om het woord te raden.



## Aantal klinkers in een zin (\*)

De gebruiker tikt een zin in. Breng op het scherm hoeveel klinkers er in de zin staan.

