

TP 1 - Go

Les TPs de ce cours peuvent être effectués sous linux/windows/macos avec l'IDE que vous souhaitez.

Si vous ne savez pas lequel choisir, l'ide recommandé est GoLand de JetBrains

Partie I : Bases

Exercice 1 : Hello world

- Compilez le code ci-dessous et exécutez-le:

```
package main

import "fmt"

func main() {
    fmt.Println("Hello, World!")
}
```

Exercice 2 : Comparaison

1. Ecrivez un programme qui demande le nom ainsi que la taille de deux personnes et affiche la chaîne suivante (dans le cas où Bob est plus grand que Marc):

Bob est plus grand que Marc

2. Modifiez le programme pour qu'il demande le genre des personnes afin d'accorder l'adjectif.

Par exemple: Alice est plus grande que Marie

3. Modifiez votre programme afin qu'il demande aussi la langue dans laquelle afficher le message.

Par exemple: Marie is taller than Bob

Exercice 3 : Boucles

On a mesuré la taille en cm de 8 personnes: 164, 208, 150, 90, 123, 156, 175, 187

1. Ecrivez un programme qui calcule la taille moyenne de ce groupe.
2. Ecrivez un programme qui affiche chaque taille dans le format suivant: 1m70
3. Modifiez votre programme afin qu'il demande à l'utilisateur de renseigner les 8 tailles lui-même.

Exercice 4: Devinettes

Créez un programme qui demande à l'utilisateur de deviner un nombre secret, le programme devra indiquer si le nombre est supérieur ou inférieur à chaque tentative ratée.

Exercice 5: Chemin

Ci dessous un cheminement qui contient des aller-retours (effectuer Gauche après Droite est inutile).

Haut, Bas, Droite, Gauche, Haut, Droite, Bas

1. Créez un programme qui affiche le même programme sans allers-retours. Note: Le chemin simplifié sera: Haut, Droite, Bas.
2. Modifiez votre programme afin qu'il demande un chemin à l'utilisateur et le simplifie.

Partie II : Avancés

Exercice 6: Affichage

- Créez un programme qui affichera les chemins de l'exercice ci-dessus dans la console.

Par exemple le chemin , Droite, Droite, Bas, Gauche, Bas, Bas, Bas, Droite, Droite, Haut sera affiché ainsi:

```
#.....
###.....
.#.....
.#.....
.#.#.....
.###.....
```

la taille du tableau (longeur/largeur) sera donnée à l'avance par l'utilisateur

- Modifiez le programme afin qu'il demande une direction à l'utilisateur et affiche le nouveau chemin.

Exercice 7: Labyrinthe

- Créez un générateur congruentiel linéaire (voir ci-dessous) afin de générer des nombres aléatoires.

Comme dans le tp 3 du cours langage C, pour générer un nombre aléatoire nous allons procéder ainsi:

- Choisir un premier nombre appelé “seed” ou “graine”.
- Appliquer l'opération $(\text{graine} * a + c) \% m$ sur ce nombre pour en produire un suivant.
- Répéter l'opération à chaque nouveau nombre.

- Générez des “murs” à des positions aléatoires dans le tableau et affichez les.

```
// les murs sont affichés avec des X
...X.....
...X....X.
...X.....
....XXX..
...X.....
.....X..
```

- Modifiez le code afin d'empêcher l'utilisateur de passer à travers les murs.

En plus de la taille du tableau (longeur/largeur) l'utilisateur renseignera le nombre total de murs

- Choisissez un point aléatoire pour le départ du labyrinthe ainsi qu'un point aléatoire pour la sortie, affichez les avec les lettres A et B.