



Exercice Fonction





Exercice 1

Faire une fonction `sqrt` qui prendra, elle aura pour prototype :

```
int sqrt(int nbrToSqrt)
```

Cette fonction prendra un nombre et le multipliera par lui-même.



Exercice 2

Faire une fonction `draw` qui prend 3 arguments et renvoie rien.

```
void draw(int height, int length, int form)
```

- 📌 form sera compris entre 0 et 3 (inclut)
- 📌 height et length corresponde a réciproquement la hauteur et la longueur de la forme a afficher

L'objectif de cette fonction est de dessiner des carrés,
Si `form` est impaire, le carré sera plein sinon vide.



Exercice 2

Si form est égale a 0 ou 1, la fonction dessinera pour:

```
draw(2, 2, 1)
```

```
o--o  
|oo|  
|oo|  
o--o
```

```
draw(1, 1, 0)
```

```
o-o  
| |  
o-o
```

```
draw(4, 2, 0)
```

```
o--o  
|  |  
|  |  
|  |  
|  |  
o--o
```



Exercice 2

Si form est égale a 2 ou 3, la fonction dessinera pour:

```
draw(2,2,3)
```

```
uxxu
```

```
xuux
```

```
xuux
```

```
uxxu
```

```
draw(1,1,2)
```

```
uxu
```

```
x x
```

```
uxu
```

```
draw(1, 5, 2)
```

```
uxxxxxxu
```

```
x      x
```

```
uxxxxxxu
```



Exercice 3

Créer un dictionnaire de `string, int` nommer phoneBook.

Créer deux fonction qui permettent d'afficher et d'ajouter des contacts.

Pour cela, créer une boucle infini qui demandera a l'utilisateur s'il veut :



ajouter un nouveau contact, ce qui appellera votre fonction `addContacts` qui prendra un nom, et un numero de téléphone.



afficher ses contacts, ce qui appellera votre fonction `listContacts`, qui affichera votre dictionnaire.



`exit`, qui arrêtera votre programme.

Évidement, un numero de téléphone sera compose uniquement de 10 chiffres, a vous de vous en assurer. Et un même numero ne pourra pas être attribué a deux personnes.