



Initiation à la Programmation C

Travaux Pratiques - L2

Fonctions et tableaux



Dans cette séance de travaux pratique, nous abordons les points suivants :

- les différents modes d'appels de fonctions ;
- l'utilisation des tableaux.

► Exercice 1. Valeur absolue

1. Écrire une fonction `valAbsolueV1` qui renvoie la valeur absolue d'un entier reçu en paramètre.

Écrire ensuite un `main` qui fasse saisir un entier à l'utilisateur, le stocke dans une variable `n`. Une nouvelle variable `p` sera alors initialisée comme la valeur de retour d'un appel à `valAbsolueV1`.

Il permettra d'obtenir la sortie suivante :

```
Saisissez un nombre : -3
Avant valAbsolueV1 : n = -3      Adresse de n = 0x7ffeebda5778
Après valAbsolueV1 : p = 3      Adresse de p = 0x7ffeebda5774
```

2. Écrire une fonction `valAbsolueV2` qui reçoit par adresse un entier et le modifie en sa valeur absolue.

Écrire ensuite un `main` qui fasse saisir en entier à l'utilisateur, le stocke dans une variable `n`, puis la modifie au cours d'un appel à `valAbsolueV2`.

Il permettra d'obtenir la sortie suivante :

```
Saisissez un nombre : -3
Avant valAbsolueV2 : n = -3      Adresse de n = 0x7ffeebda5778
Après valAbsolueV2 : n = 3      Adresse de n = 0x7ffeebda5778
```

► Exercice 2. Equation du second degré

Le calcul de la racine carrée s'effectue avec la fonction `sqrt` de la bibliothèque mathématique. Pour pouvoir l'utiliser, il faut :

- inclure le fichier d'en-tête `math.h` avec la ligne `#include <math.h>`
- compiler avec l'option `-lm` :

```
clang -std=c17 monProg.c -o monProg -lm
```

1. Compléter le code de la fonction `resoudre` disponible dans le fichier `EqSecondDegre.c`
2. Écrire une fonction `main` permettant de chercher et afficher les racines de dix équations du second degré. Les coefficients seront tous demandé à l'utilisateur.

Bien sur, nous n'avons pas envie d'entrer manuellement trente coefficients pour tester notre programme. Pour cela, nous allons utiliser les redirections du `shell` en dérivant l'entrée standard vers le fichier `polynomes.txt` à l'aide de la commande `monprog < fichier` : la lecture avec `scanf` se fera ainsi dans le fichier `fichier` et non pas via l'entrée standard.

► Exercice 3. Tableau

1. Écrire une fonction `void afficheTab(int tab[], int taille)` qui affiche les `taille` premiers éléments du tableau `tab` sur la sortie standard.
2. Définir dans la fonction `main` un tableau contenant les entiers de 1 à 5 et l'afficher.
3. Écrire une fonction `void changeTab(int tab[], int taille int indice)` qui ajoute 2 à l'élément dans la case `indice` du tableau `tab`. *On supposera que `indice` est un indice valide du tableau `tab`.*
4. Modifier la fonction `main` pour que celle-ci appelle la fonction `changeTab` après son affichage, puis le réaffiche.

Pour expliquer la transformation réalisée par `changeTab`, afficher les valeurs des adresses de `tab` et `&tab[0]` dans le `main` ainsi que dans la fonction `changeTab`.