
TD INFORMATIQUE 05 : BRANCHEMENTS CONDITIONNELS

Consignes générales : N'oubliez pas pour ce TD comme pour les suivants de vous créer un répertoire consacré au TD et d'enregistrer vos codes dessus.

On rappelle que les commandes à taper dans le terminal pour compiler puis exécuter votre programme C :

— Pour compiler : `gcc -o nom_executable nom_programme.c`

— Pour exécuter : `./nom_executable`

Il est conseillé de toujours écrire l'algorithme en pseudo-code avant de passer sur machine.

Exercice 1

Soit l'algorithme suivant :

```
PROGRAMME test condition
VARIABLES
  a, b : entier
  c1 , c2 : booléen
DEBUT
  ÉCRIRE("Saisir deux entier")    LIRE(a)
  LIRE(b)
  c1 ← a STRCTEMENT SUPERIEUR A b
  c2 ← a INFÉRIEUR OU ÉGAL A 0
  SI (c1) ALORS
    ÉCRIRE (a*b)
  SINON
    ÉCRIRE (a+b)
  FIN SI
  SI (c2) ALORS
    ÉCRIRE(b-a)
  SINON SI ( a ÉGAL b)
    ÉCRIRE("même nombres !")
  SINON
    ÉCRIRE ("a"+a "b"+b)
  FIN SI
FIN
```

Que va afficher l'algorithme si l'utilisateur saisi

- 5 puis 2
- 10 puis 10
- -4 puis -4
- 15 puis -15

Exercice 2

Que va afficher l'algorithme suivant ?

```
PROGRAMME carte
VARIABLES
  x, y : entier
  z : réel
DEBUT
  x← 10
  y← 3
  z← x/y
  SI (z ≤ 3) FAIRE
    ÉCRIRE ("Carreau")
    y← 5
  FIN SI
  z←x-y
  SI (z>4) FAIRE
    ÉCRIRE("Pique")
```

```

SINON SI (x>5)
    ÉCRIRE("Coeur")
SINON
    ÉCRIRE ("Trèfle")
FIN SI
SI(x>5 ET y<5)
    ÉCRIRE("As")
FIN SI
FIN

```

Exercice 3

Un étudiant étourdi a traduit l'algorithme de l'exercice 1 en C mais a commis des erreurs. Corriger les erreurs et tester sur machine.

```
#include <stdio.h>
```

```

int main(){
    int a, b;
    printf("Saisir deux entiers\n");
    scanf ("%f",&a);
    scanf ("%f",&b);
    if (a<b){
        printf("a*b");

    else
        printf("a+b");
    }
    if(a<=0){
        printf("b-a");
    }
    else if(a=b){
        printf("Meme nombre!");
    }
    else{
        printf("a%d b%d",a,b);
    }

    return ;
}

```

Exercice 4

Écrire un programme qui saisit un entier et affiche sa valeur absolue.

Exercice 5

Écrire un programme qui saisit un entier et affiche s'il est pair ou impair.

Exercice 6 (*mention d'une note*)

1. Demander à l'utilisateur de donner une note entre 0 et 20.
2. Afficher un message en fonction de la note :
 - "Ajourné" si la note est inférieure ou égale à 6.
 - "Rattrapage" si la valeur est supérieure à 6 et inférieure à 10.
 - "Admis" si la valeur est supérieure ou égale à 10.
 - Affichez un message d'erreur si la note est totalement en dehors de l'intervalle [0,20].

Exercice 7 (*valeur maximum*) A CY-Tech, on est sympa ! Nous avons instauré la règle de la seconde chance : sur les N notes en informatique du semestre, seules les N-1 meilleures sont utilisées pour faire la moyenne. Écrire un algorithme qui affiche la moyenne à partir de 3 notes saisies en respectant cette règle.

Exercice 8 (*Signe d'une multiplication*)

Attention : vous n'avez pas le droit d'utiliser l'opérateur multiplication dans cet exercice !

1. Écrire un algorithme qui va demander à l'utilisateur de saisir 2 valeurs entières
2. Déterminer si le produit de ces 2 variables est positif, négatif ou nul et afficher cette information.

Exercice 9 (*Année bissextile*) Une année est bissextile (elle comporte 366 jours) si elle est divisible par 4 et non divisible par 100 ou si elle est divisible par 400. Écrire un programme qui permet de déterminer si une année saisie par l'utilisateur est bissextile.

Exercice 10 (*Dans un cercle ?*) Pour cet exercice on pourra utiliser la commande `sqrt()` qui permet de calculer la racine carrée d'un nombre. Cette commande est dans la bibliothèque `math.h` qu'il faudra donc inclure. On va déterminer si un point est dans un cercle.

1. On considère un cercle donc les coordonnées du centre sont $(0,0)$ et son rayon un réel choisi par l'utilisateur.
2. Saisir les coordonnées d'un point x et afficher si ce point appartient au cercle, et en dehors du cercle, ou est sur le cercle.
3. Modifier l'algorithme précédent pour que l'utilisateur puisse également choisir les coordonnées du centre du cercle.

Exercice 11 (*Nature d'un triangle*)

1. Demander à l'utilisateur de saisir les longueurs des côtés d'un triangle en commençant par la plus grande.
2. Déterminer si ce triangle est équilatéral, isocèle, rectangle, rectangle isocèle ou quelconque (on rappelle l'existence de la commande `sqrt`).

Exercice 12 (*Vente en gros*) Soit un magasin qui vend des masques à l'unité avec une offre promotionnelle :

- Les 20 premiers masques achetés coutent 20 cts chacun.
- Les 10 masques suivants coûtent 15 cts chacun.
- Tous les masques supplémentaires seront facturés 10 cts.

Écrivez un algorithme qui demande à l'utilisateur le nombre de masques dont il a besoin et qui affichera le prix correspondant.

Exercice 13 (*Calculatrice*) Nous allons simuler le fonctionnement d'une calculatrice.

- Demander à l'utilisateur de saisir deux nombres réels a et b qui seront les opérandes de l'opération à effectuer.
- Demander à l'utilisateur de saisir l'opérateur souhaité ($+$, $-$, $/$ ou \times). Afficher le résultat de l'opération ou bien un message d'erreur si la donnée saisie ne correspond pas à un opérateur. (Attention! La donnée saisie est ici un caractère, pas un opérateur, on ne peut pas l'utiliser tel quel!).