Université de Marne-la-Vallée

Travaux Dirigés d'algorithmique n°2

Cours d'Informatique de Première Année

—DEUG Sciences S2-SM/STPI—

Variables, instructions conditionnelles

Dans cette séance, nous abordons les points suivants :

- les entrées/sorties en C ;
- les types de base ;
- les structures conditionnelles *if* et *if . . . else* ;
- la structure de sélection à choix multiples *switch*.

▸ **Exercice 1.** ***(Lecture au clavier)***
La lecture au clavier se fait grace à la fonction `scanf`*. Il faut lui indiquer exactement le type des données à lire :*
`%d` *pour* `int`, `%f` *pour* `float`, `%c` *pour* `char`*.*
(Pour les deux premiers types, les espaces, tabulations et passages à la ligne sont considérés comme des séparateurs.)
Chaque variable à lire doit être précédée du symbole `&`*.*
Ainsi “ `scanf("%d%f%d",&i,&x,&j);` *” lit successivement au clavier un entier affecté à la variable* `i`*, un réel affecté à la variable* `x` *et un entier affecté à la variable* `j`*. Comme pour* `printf`*, il doit y avoir concordance entre les types et le nombre des valeurs à lire, et les variables fournies à la fonction.*

- *Écrire un programme qui demande un entier à l'utilisateur, le lit au clavier puis l'affiche entre deux lignes vides.*
- *Écrire un programme utilisant deux variables qui affiche la somme de deux entiers saisis au clavier. (même question pour la différence, le produit et le quotient de la division euclidenne).*
  Modifier votre programme pour que la division ne soit effectuée que si la seconde variable est non nulle.
- *On souhaite maintenant réaliser une mini-calculette. Écrire un programme qui lit deux entiers et un caractère, et qui réalise l'opération associée au caractère saisi (l'addition pour* `+`*, la soustraction pour* `-`*, la multiplication pour* `*`*, la division entière pour* `/`*.*
  Pour cela, vous utiliserez la structure de sélection `switch` *dont la syntaxe est la suivante :*

```
switch(expression)
{
  case val_1 : instruction_1; break;
  case val_2 : instruction_2; break;
  .... ..... : ............
  case val_k : instruction_k; break;
  default    : instruction_par_defaut;
}
```

▸ **Exercice 2.** ***(Structure conditionnelle)***
Écrire un programme C qui lit un nombre entier saisi au clavier et qui indique si ce nombre est divisible par 2 et/ou par 3. Nous souhaitons un affichage du type suivant :
`divisible par 2`
ou `divisible par 3`
ou `divisible par 2 et par 3.`

▸ **Exercice 3.** ***(Étude du nombre de solutions d'une équation du second degré)***
On veut déterminer le nombre de solutions d'une équation du second degré $ax^2 + bx + c = 0$. Nous envisagerons les cas suivants :

- *si $a = 0$ et $b = 0$, l'équation est dégénérée (0 ou une infinité de solutions) ;*
- *si $a = 0$ et $b \neq 0$, il y a une racine ;*
- *si $a \neq 0$ et $c = 0$, il y a deux racines ;*
- *sinon, on utilise le discriminant $b^2 - 4ac$ pour déterminer le nombre de solutions :*
  - *si le déterminant est négatif, pas de racine réelle ;*
  - *si le déterminant est nul, une racine double ;*
  - *si le déterminant est positif, deux racines.*

Écrire un programme qui lit les coefficients `a`, `b` *et* `c`, *et qui calcule le nombre de solutions de l'équation du second degré associée.*
Modifier le programme pour qu'il calcule les solutions (quand elles existent).

▸ **Exercice 4.** ***(Équations de droites dans le plan)***

- *Décrire l'algorithme permettant de déterminer les coefficients a et b de l'équation $y = a\,x + b$ d'une droite du plan à partir de deux points ;*
- *écrire un programme qui affiche l'équation de la droite passant par deux points P et Q dont les coordonnées auront été saisies au clavier ;*
- *décrire l'algorithme permettant de déterminer le point d'intersection de deux droites ;*
- *transformer le programme précédent pour afficher le point d'intersection de deux droites.*