

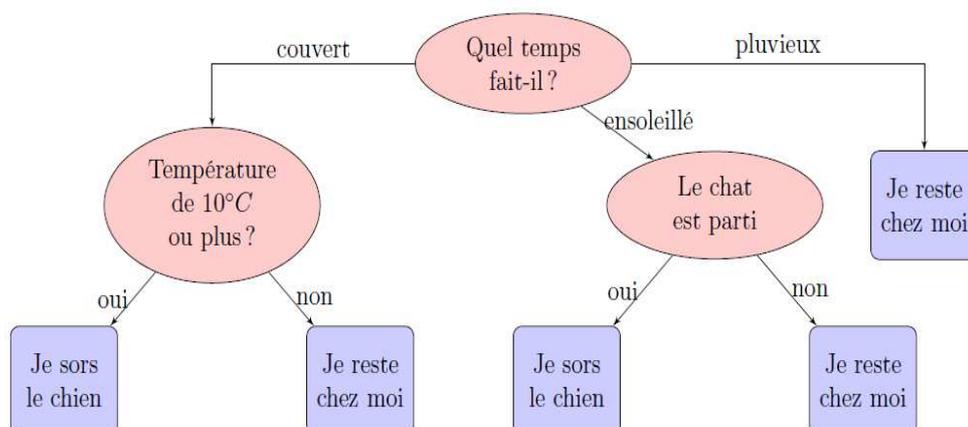
Travaux pratiques 2 Programmation en Langage C

Exercice 1:

- Soient 3 variables a, b, c, initialisées à des valeurs quelconques.
 - Écrire un programme qui calcule et affiche à l'écran le minimum des 3 valeurs.
 - Écrire un programme qui calcule et affiche à l'écran le maximum des 3 valeurs.
- Considérant maintenant un nombre quelconque de variable spécifié par l'utilisateur
 - Écrire un programme qui calcule et affiche à l'écran le minimum des valeurs.
 - Écrire un programme qui calcule et affiche à l'écran le maximum des valeurs.
 - Écrire un programme qui calcule et affiche à l'écran la somme des valeurs
 - Écrire un programme qui calcule et affiche à l'écran le produit des valeurs
 - Écrire un programme qui calcule et affiche à l'écran la moyenne générale des valeurs.
- Reprendre la question 2 et écrire un programme qui réalise une opération de choix

Exercice 2:

Un arbre de décision est un graphe particulier où les nœuds sont des questions et les arêtes sont les réponses à ces questions. Il se lit de haut en bas. On avance dans l'arbre en répondant aux questions. Les nœuds les plus bas jouent le rôle particulier de classes de réponse au problème initial.



Écrire un programme implantant l'arbre de décision au dessus pour proposer une réponse étant donné un jour. Après chaque test effectué sur une variable, afficher sa valeur afin de suivre la progression dans l'arbre.

Exercice 3:

Étant donnée une heure représentée sous la forme de 3 variables pour les heures, h, les minutes, m et les secondes, s

- a. Écrire un programme qui affiche l'heure qu'il sera 1 seconde plus tard selon les deux formats (12h et 24h).
- b. Même question pour n secondes plus tard (n est saisi au clavier).

Exercice 4:

1. Saisir les coefficients a et b et afficher la solution de l'équation $ax + b = 0$.
2. Saisir les coefficients a, b et c, afficher la solution de l'équation $ax^2 + bx + c = 0$.
3. Etant donnés les réels $a_1, a_2, b_1, B_2, c_1, c_2$. Imprimer la solution du système d'équations
$$\begin{aligned} a_1 * x + b_1 * y &= c_1 \\ a_2 * x + b_2 * y &= c_2 \end{aligned}$$
4. Etant données trois longueurs réelles a, b et c, concevoir un algorithme qui détermine si a, b et c peuvent être les cotés d'un triangle et, si oui, si le triangle en question est rectangulaire, isocèle ou équilatéral. On veut aussi, dans le cas où le triangle n'est pas rectangle s'il est aigu ou obtus.

Exercice 5:

Un nombre parfait est un entier positif strict qui est égal à la somme de ses diviseurs. Le premier nombre parfait est $6 = 1 + 2 + 3$. L'algorithme doit imprimer tous les nombres parfaits entre 1 et n ainsi que leur nombre. Par exemple, pour $n = 8$, il doit imprimer 6 et 1.

Exercice 6:

Afficher les premiers termes de la suite U_n inférieurs à 500. Puis afficher le premier rang du terme supérieur à un entier valide saisi au clavier.

$$U_0 = 4$$

$$U_1 = 2$$

$$U_n = 3 * U_{n-1} + 4 * U_{n-2}$$

Exercice 7:

- 1: Écrire un programme qui, étant données deux variables, longueur et largeur, initialisées à des valeurs strictement positives quelconques, affiche un rectangle d'étoiles correspondant
- 2: Écrire un programme qui affiche, étant donnée la variable, cote, initialisée à une valeur quelconque, un demi-carré d'étoiles (triangle rectangle isocèle) ayant cote comme longueur de côté
- 3: Écrire un programme qui affiche une pyramide d'étoiles de base et hauteur spécifiés par l'utilisateur.
- 4: Écrire un programme qui affiche une grande étoile composée d'étoiles