

Algorithmique - TP1

Mise en route: utilisation de la console

Étudiez, compilez et exécutez le programme `Salutations` fourni.

Complétez le programme pour qu'il demande l'âge de l'utilisateur, puis affiche l'âge qu'il aura dans 20 ans.

Affectations, type des données

Étudiez le programme `Affectations` fourni. Sans exécuter le programme, notez dans le tableau suivant les valeurs affichées par ce programme.

| | a | b | c | d | e |
|--------|---|---|---|---|---|
| Step 1 | | | | | |
| Step 2 | | | | | |
| Step 3 | | | | | |

Compilez et exécutez le programme (après d'éventuelles corrections). Comparez les résultats affichés avec ceux que vous aviez prévus.

Complétez le programme pour qu'il transfère à `a` la valeur de `b` et à `b` la valeur de `a`.

Ajoutez une troisième variable entière `x`, et complétez le programme pour qu'il transfère à `a` la valeur de `b`, à `b` la valeur de `x` et à `x` la valeur de `a`.

Conditionnelles: dates

Écrivez un programme `Date` qui demande la date de naissance de l'utilisateur sous forme de trois entiers `day`, `month` et `year` et les affiche sous la forme `day/month/year`.

Compléter le programme pour qu'il vérifie que $1 \leq \text{day} \leq 31$, $1 \leq \text{month} \leq 12$ et $1900 \leq \text{year} \leq 2100$ avant d'afficher la date.

Compléter le programme pour qu'il vérifie que la date est valide (penser au nombre de jours de chaque mois et aux années bissextiles)

Itérations: suites d'entiers

Écrivez un programme `Sequence` qui lit une suite de 10 entiers positifs et affiche la moyenne de ces entiers en utilisant une boucle `for`.

Réécrivez ce programme avec une boucle `while`. Laquelle de ces deux boucle est la plus adaptée?

Modifiez ce programme pour qu'il lise une suite d'entiers positifs terminée par `-1` (on ne connaît pas à l'avance le nombre d'entiers saisis) et affiche la moyenne de ces entiers, en utilisant une boucle `while`.

Réécrivez ce programme avec une boucle `do . . . while`. Laquelle de ces deux boucle est la plus adaptée?

Complétez ce programme pour qu'il affiche le maximum et le minimum de la suite.

Complétez ce programme pour qu'il affiche le nombre d'occurrences du premier entier de la suite.

Complétez ce programme pour qu'il affiche le nombre de monotonies de la suite: une monotonie est une suite croissante d'entiers (exemple: la suite 12 4 16 4 4 8 12 contient 4 monotonies: (12) (4 16) (4) (4 8 12)).

Modifiez ce programme pour qu'il lise la suite d'entiers, puis affiche le menu affiché ci dessous, lise le choix de l'utilisateur, et l'exécute, jusqu'à ce que l'utilisateur sélectionne 'q', en choisissant le type de boucle (for, while ou do ... while la plus adaptée). Le menu proposé:

- a. afficher la moyenne
- b. afficher le minimum
- c. afficher le maximum
- d. afficher le nombre d'occurrences du 1er entier
- e. afficher le nombre de monotonies
- q. quitter

Dessins

Écrire un programme qui, à l'aide du signe '*', affiche un carré de côté n (n est donné par l'utilisateur):

```
* * *
* * *
* * *
```

Variantes:

```
*           *           *           * * *           * *
* *         * * *       * * *       * * *           * *
* * *       * * * * *   * * * * *   * * *           * *
                        * * *
                        *
```

Manipulations sur un nombre entier

Écrire un programme qui demande à l'utilisateur un nombre entier, et calcule l'entier correspondant à ce nombre inversé (exemple: 5314 devient 4135).

Complétez ce programme pour qu'il vérifie si ce nombre est premier.

Complétez ce programme pour qu'il donne la décomposition du nombre en facteurs premiers (exemple: 45 donne 3 3 5).

Complétez ce programme pour qu'il demande un deuxième entier et calcule le PGCD des deux entiers en utilisant la méthode d'Euclide (méthode par divisions successives), fondée sur la propriété suivante:

- si b est un diviseur de a, alors $\text{PGCD}(a, b) = b$
- pour $a > b$, $\text{PGCD}(a, b) = \text{PGCD}(b, a \% b)$

Dates (suite)

Compléter/modifiez/restructurez le programme pour qu'il demande une deuxième date, la date du jour, la vérifie et l'affiche.

Compléter le programme pour qu'il affiche en plus la date de demain.

Compléter le programme pour qu'il affiche le nombre de jours entre les deux dates.

Compléter le programme pour qu'il affiche l'âge de l'utilisateur.

Manipulations sur un nombre entier (suite)

Question subsidiaire : l'algorithme d'Euclide étendu : le théorème de Bézout établit que le PGCD d de deux entiers a et b est une combinaison linéaire (à coefficients entiers) de a et b , i.e. $d = au + bv$.

Une modification simple de l'algorithme d'Euclide (qu'on appelle alors algorithme d'Euclide *étendu*) permet de calculer ces coefficients u et v . Remarquons tout d'abord que l'algorithme d'Euclide calcule une suite d'entiers définie par une récurrence à deux termes :

$$a_0 = a, a_1 = b$$

$$a_{n-1} = q_n a_n + a_{n+1}$$

$$\text{autrement dit : } a_{n+1} = -q_n a_n + a_{n-1}$$

Si l'on pose les récurrences : $u_{n+1} = -q_n u_n + u_{n-1}$ et $v_{n+1} = -q_n v_n + v_{n-1}$ avec les conditions initiales suivantes :

$$u_0 = 1, v_0 = 0$$

$$u_1 = 0, v_1 = 1$$

I) Montrer par récurrence que, pour tout n , $a_n = a u_n + b v_n$.

Dès que $a_{n+1}=0$, on obtient ainsi $a_n = \text{PGCD}(a, b)$ avec u_n et v_n les coefficients entiers de l'identité de Bézout.

Exemple : $a=96, b=81$

| n | a_n | a | u_n | b | v_n |
|-----|-------------------|----------|------------------|----------|------------------|
| 0 | 96 | $= 96 *$ | 1 | $+ 81 *$ | 0 |
| 1 | 81 | $= 96 *$ | 0 | $+ 81 *$ | 1 |
| 2 | $15 = 96 - 81$ | | | | |
| | | $= 96 *$ | $(1 - 0)$ | $+ 81 *$ | $(0 - 1)$ |
| | | $= 96 *$ | 1 | $+ 81 *$ | -1 |
| 3 | $6 = 81 - 5 * 15$ | | | | |
| | | $= 96 *$ | $(0 - 5 * 1)$ | $+ 81 *$ | $(1 - 5 * (-1))$ |
| | | $= 96 *$ | -5 | $+ 81 *$ | 6 |
| 4 | $3 = 15 - 2 * 6$ | | | | |
| | | $= 96 *$ | $(1 - 2 * (-5))$ | $+ 81 *$ | $(-1 - 2 * 6)$ |
| | | $= 96 *$ | 11 | $+ 81 *$ | -13 |

II) Réalisez une fonction qui calcule le PGCD de deux nombres a et b donnés en entrée ainsi que les coefficients de l'identité de Bézout.