Travaux Dirigés n°2

Rappel : Pour chaque problème, il vous est demandé de définir clairement :

- les données d'entrée du problème en précisant leurs types (nombre entier, réel, ...) ;
- les éventuelles données de sortie du problème en précisant leurs types ;
- les instructions permettant d'obtenir les données de sortie à partir des données d'entrée.

Tester ensuite votre algorithme à la main à partir de données d'entrées judicieusement choisies pour explorer les différents cas de fonctionnement.

Exercice 1

Inversion de variables

Écrire un algorithme qui prend en entrée deux variables entières `a` et `b` et qui inverse leurs contenus (si possible sans utiliser de variable locale).

Solution

Données d'entrée : a et b entiers
Données de sortie : inverse les contenus
1: $a \leftarrow a + b$
2: $b \leftarrow a - b$
3: $a \leftarrow a - b$

Exercice 2

Algorithme d'Euclide

Écrire un algorithme qui calcule le `PGCD` de deux nombres entiers strictement positifs en utilisant l'algorithme d'Euclide.

On rappelle le principe à la base de cet algorithme :

$$\texttt{PGCD}(a, b) = \texttt{PGCD}(b, a \mod b) \quad \text{et} \quad \texttt{PGCD}(a, 0) = a$$

Solution

Données d'entrée : a et b entiers strictement positifs
Données de sortie : $pgcd$ le `PGCD` de ces deux nombres

Si $a < b$ **Alors**
  Inverser a et b
Fin Si
Tant que $b \neq 0$ **Faire**
  $a \leftarrow a \mod b$
  Inverser a et b
Fin Tant que
$pcgd \leftarrow a$
Renvoyer $pgcd$

Exercice 3

Fractions égyptiennes

Toute fraction peut s'écrire comme une somme de fractions ayant 1 comme numérateur. Cette décomposition est appelée décomposition en fractions égyptiennes. En voici un exemple :

$$\frac{7}{8} = \frac{1}{2} + \frac{1}{3} + \frac{1}{24}$$

Écrire un algorithme prenant en entrée le numérateur n et le dénominateur d d'une fraction et affiche sa décomposition en fractions égyptiennes.

Solution

Données d'entrée : n et d entiers, numérateur et dénominateur de la fraction $\frac{n}{d}$
Données de sortie : affiche la décomposition en fractions égyptiennes
Données locales : variable de boucle i.

$i \leftarrow 2$
Tant que $i \cdot n \neq d$ **Faire**
  Si $i \cdot n \geq d$ **Alors**
    Afficher " $\frac{1}{i}+$ "
    $n \leftarrow i \cdot n - d$
    $d \leftarrow i \cdot d$
  Fin Si
  $i \leftarrow i + 1$
Fin Tant que
Afficher " $\frac{1}{i}$ "

Exercice 4

Calcul de la racine carrée

Cet exercice consiste à regarder plusieurs techniques pour le calcul de la racine carrée d'un nombre. Dans un premier temps on cherchera à calculer la racine carrée entière d'un nombre c'est-à-dire la troncature de la racine carrée de ce nombre. Puis on cherchera à déterminer la racine d'un nombre réel quelconque.

1. Donner un algorithme simple (dit naïf) qui prend un entier a en entrée et calcule sa racine carrée entière *racine*.

$$racine = \max\{n \in \mathbb{N}, n^2 \leq a\}$$

2. Donner un algorithme qui prend un entier a en entrée et calcule sa racine carrée entière *racine* en utilisant une méthode alternative basée sur la remarque suivante :

$$\sum_{i=1}^{p} 2i - 1 \leq n < \sum_{i=1}^{p+1} 2i - 1 \quad \Longleftrightarrow \quad p \leq \sqrt{n} < p + 1$$

3. Donner un algorithme qui prend en entrée un nombre réel a et renvoie une valeur approchée de sa racine carrée en utilisant la méthode de Newton exposée ci-après :
La suite r_n converge vers la racine de a

$$\begin{cases} r_0 = 1, \\ r_{n+1} = \dfrac{r_n + \frac{a}{r_n}}{2}. \end{cases}$$

Solution

Données d'entrée : Un nombre entier a
Données de sortie : La racine carrée entière *racine* de a

$racine \leftarrow 0$
Tant que $racine \cdot racine \leq a$ **Faire**
  $racine \leftarrow racine + 1$
Fin Tant que
Renvoyer $racine - 1$

Solution

Données d'entrée : Un nombre entier a
Données de sortie : La racine carrée entière *racine* de a
Données locales : un entier *somme*.

$racine \leftarrow 0$
$somme \leftarrow 0$
Tant que $somme \leq a$ **Faire**
  $racine \leftarrow racine + 1$
  $somme \leftarrow somme + 2 \cdot racine - 1$
Fin Tant que
Renvoyer $racine - 1$

Solution

Données d'entrée : Un nombre réel a et un réel p qui donne la précision voulue : $|racine^2 - a| \leq p$
Données de sortie : Un réel *racine*, valeur approchée de la racine carrée de a

$racine \leftarrow 1$
Tant que $|racine^2 - a| > p$ **Faire**
  $racine \leftarrow \dfrac{racine + \frac{a}{racine}}{2}$
Fin Tant que
Renvoyer $racine$

Exercice 5

Fractions continues

On va maintenant s'intéresser au développement en fractions continues des nombres réels. Un développement en fractions continues d'un nombre réel a est une suite d'entiers $(a_0, \ldots, a_n)$ finie ou non, avec $a_1, \ldots, a_n > 0$, telle que :

$$a = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \ldots}}$$

Donner un algorithme qui permet de calculer un tel développement.

Solution

Données d'entrée : Un réel a
Données de sortie : Affiche un développement en fractions continues
Données locales : un entier *decomp*.

```
Tant que a ≠ 0 Faire
    decomp ← ⌊a⌋
    a ← 1 / (a − decomp)
    Afficher decomp
Fin Tant que
```