

# Racine carrée entière [it03]

## Exercice résolu

Karine Zampieri, Stéphane Rivière

Unisciel  algoprog  Version 17 mai 2018

## Table des matières

<b>1</b>	<b>Racine carrée entière / pgracine</b>	<b>2</b>
1.1	Spécification et invariants	2
1.2	Fonction racine (racine entière)	3
1.3	Programme de test	3
<b>2</b>	<b>Références générales</b>	<b>4</b>

## Java - Racine carrée entière (Solution)



Mots-Clés Schéma itératif ■

Utilise Spécification, Déduction du programme ■

Difficulté ●●○ (20 min) ■



### Objectif

Cet exercice calcule la racine carrée entière d'un entier positif ou nul.

# 1 Racine carrée entière / pgracine

## 1.1 Spécification et invariants

Voici un exemple du résultat attendu :

```
Un entier? 28
Racine carrée entière de 28 vaut 5
```

En effet  $25 = 5^2 \leq 28 < 6^2 = 36$ .

On recherche donc la racine carrée entière  $i$  d'un entier  $a$  positif ou nul :

$$i^2 \leq a < (i + 1)^2 \quad a \in \mathbb{N}, r \in \mathbb{N}$$

Et pour le calcul de la puissance, l'algorithme ne doit utiliser **que les opérations additives** et non la multiplication.

Considérons alors la suite  $y = (y_i)$  dont la définition directe est :

$$\begin{cases} y_i = i^2 \\ i^* = PPETIT(i \geq 0 : y_{i+1} > a) \end{cases}$$



Recherchez une définition récurrente d'ordre 1.

### Solution simple

Exprimons  $y_{i+1}$  en fonction de  $y_i$  :

$$\begin{aligned} y_{i+1} &= (i + 1)^2 \\ &= i^2 + 2i + 1 \\ &= y_i + z_i \end{aligned}$$

en posant  $z_i = 2i + 1$ . On exprime alors à nouveau  $z_{i+1}$  en fonction de  $z_i$  :

$$\begin{aligned} z_{i+1} &= 2(i + 1) + 1 \\ &= 2i + 2 + 1 \\ &= z_i + 2 \end{aligned}$$

Le calcul de  $y_{i+1}$  se ramène donc au calcul des deux suites :

$$\begin{cases} y_{i+1} = y_i + z_i \\ z_{i+1} = z_i + 2 \end{cases}$$



Nous avons les relations suivantes, quel que soit  $i$  :

$$\begin{cases} y_i = i^2 \\ z_i = 2i + 1 \end{cases}$$

En introduisant une suite  $(u_i)$  telle que  $u_i = i$ , exprimez  $u_i$  en fonction de  $z_i$ .

**Solution simple**

Comme  $u_i = i$  ou encore  $u_i = u_{i-1} + 1$ , il vient :

$$\begin{cases} y_i = u_i^2 \\ z_i = 2u_i + 1 \end{cases}$$

Ces deux relations sont vraies avant et après l'étape d'itération  $i$ . Elles nous permettent d'exprimer  $u_i$  en fonction de  $z_i$  :

$$u_i = (z_i - 1) \text{ div } 2$$

**1.2 Fonction racine (racine entière)****Propriété**

Le **SDR** de la racine carrée entière  $i$  d'un entier  $a$  positif ou nul s'écrit :

$$\begin{aligned} i^* &= \text{PPETIT}(i \geq 0 : y_{i+1} > a) \\ &= u^* - 1 \\ &= (z^* - 3) \text{ div } 2 \end{aligned}$$



On introduit les variables **y**, **z** et **u** pour représenter chacune des suites  $(y_i)$ ,  $(z_i)$  et  $(u_i)$ .  
Donnez le graphe de précedence ainsi que les initialisations.

**Solution simple**

Le graphe de précedence est  $y \mapsto z$  : il faut calculer **y** en premier.  
A l'initialisation,  $y_0 = 0$  donc  $z_0 = 1$  et  $u_0 = 0$ .



Déduisez une fonction `racine(a)` qui calcule et renvoie la racine carrée d'un entier **a**.



Donnez le nombre d'additions et justifiez la terminaison.

**Solution simple**

On a :

- Nombre d'additions : il y en a  $3 \times u$ .
- Terminaison : La suite des  $(y_i)$  est croissante : elle finit nécessairement par dépasser  $a$  qui est une valeur fixe.

**1.3 Programme de test**

Écrivez un programme afin de tester votre fonction.



Testez.



Validez votre fonction et programme avec la solution.

**Solution Java** @[pgracine.java]

```
import java.util.Scanner;

class PGRacine {

    /**
     * Racine carrée entière
     * @param[in] a - un entier
     * @return la racine carrée entière de a
     */

    public static int racine(int a)
    {
        int y = 0, z = 1, u = 0;
        while (y <= a)
        {
            y += z;
            z += 2;
            u += 1;
        }
        return (u - 1);
    }

    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        int a;
        System.out.print("Un entier? ");
        a = input.nextInt();
        System.out.println("Racine carree entiere de " + a + " vaut " + racine(a));
    }
}
```

## 2 Références générales

Comprend [] ■