Approximation du nombre π

Algo & Prog avec R

A. Malapert, B. Martin, M. Pelleau, et J.-P. Roy

9 avril 2019

Université Côte d'Azur, CNRS, I3S, France
`firstname.lastname@univ-cotedazur.fr`

Le nombre π

- π est défini comme le rapport constant entre la circonférence d'un cercle et son diamètre dans le plan euclidien.
- De nos jours, les mathématiciens définissent π par l'analyse réelle à l'aide des fonctions trigonométriques elles-mêmes introduites sans référence à la géométrie.
- Le nombre π est irrationnel, ce qui signifie qu'on ne peut pas l'écrire comme une fraction.
- Le nombre π est transcendant ce qui signifie qu' il n'existe pas de polynôme à coefficients rationnels dont π soit une racine.

Calcul de π par la formule de Leibniz

On utilisera la formule de Leibniz issue du développement en série de Taylor en 0 de $arctan(x)$ évalué au point 1 :

$$\sum_{k=0}^{+\infty} \frac{(-1)^k}{2k+1} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{9} + \frac{1}{11} + \ldots = \frac{\pi}{4}$$

Elle a été découverte en Occident au XVIIe, mais apparaît déjà chez Madhava, mathématicien indien de la province du Kerala, vers 1400.

c.f. Wikipedia

Nous allons développer un algorithme d'approximation de π.

ITÉRATION Comment améliorer l'approximation courante ?

TERMINAISON Mon approximation courante est-elle assez bonne ?
Est-ce que le calcul prend trop de temps ?

INITIALISATION Comment initialiser la première approximation ?

Algorithme d’approximation de π

ITÉRATION

Pour améliorer l’approximation, étant en possession de la somme acc des i premiers termes, on voudra obtenir la somme des $i+1$ premiers. Il su□ra donc d’incrémenter i, puis d’ajouter $\frac{(-1)^i}{2i+1}$ à acc.

```
i <- i + 1
term <- (-1)**i / (2*i + 1)
acc <- acc + term
```

TERMINAISON

Mon approximation courante a est-elle assez bonne ? Elle est assez bonne lorsque je n’arrive plus à l’améliorer. Notons h la précision.

Est-ce que le calcul prend trop de temps ? Notons n le nombre maximum de termes à calculer.

```
abs(term) < h || i > n
```

INITIALISATION

```
i <- 0
acc <- 1
```

Programme d'approximation de π

```r
LeibnizPi <- function(n = 10**4, h = 2^(-20)) {
  i <- 0
  term <- 1
  acc <- 1
  while( (i <= n) && 4*abs(term) > h) {
    i <- i + 1
    term <- (-1)**i / (2*i + 1)
    acc <- acc + term
  }
  return(4*acc)
}
```

```
> LeibnizPi(n = 100, h = 0)
[1] 3.131789
> LeibnizPi(n = 100000, h = 0)
[1] 3.141583
> pi
[1] 3.141593
```