TP3 Simulation

Généré par Doxygen 1.8.13

Table des matières

1 Index des fichiers **1**

1.1 Liste des fichiers 1

2 Documentation des fichiers **3**

2.1 Référence du fichier main.c 3

2.1.1 Description détaillée 4

2.1.2 Documentation des fonctions 4

2.1.2.1 calc_radius() 4

2.1.2.2 calc_variance() 5

2.1.2.3 fibo() 5

2.1.2.4 get_t() 5

2.1.2.5 monte_carlo() 6

2.1.2.6 replicates_monte_carlo() 6

2.1.3 Documentation des variables 7

2.1.3.1 t_values 7

Index **9**

Chapitre 1

Index des fichiers

1.1 Liste des fichiers

Liste de tous les fichiers documentés avec une brève description :

main.c 3
mt19937ar.h . . . . . **??**

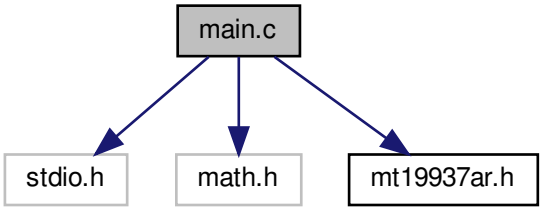
Chapitre 2

Documentation des fichiers

2.1 Référence du fichier main.c

```
#include <stdio.h>
#include <math.h>
#include "mt19937ar.h"
```

Graphe des dépendances par inclusion de main.c :



Macros

- #define NB_VALUES 100
  Nombre de valeur de utilisée pour faire les statistiques.

Fonctions

- double monte_carlo (int nb_points)
  Calcule la valeur de PI en utilisant la méthode de Monte-Carlo.
- double replicates_monte_carlo (int nb_replicates, double ∗values)
  Réitère la méthode de Monte-Carlo pour en faire des statistiques.
- double calc_variance (int nb, double ∗vals, double mean)
  Calcule de la variance du jeu de valeur.
- double get_t (int nb)
  Permet d'obtenir le quantile correspondant au nombre de valeurs que l'on utilisent.
- double calc_radius (int nb, double ∗vals, double variance)
  Calcule le rayon de l'intervalle de confiance.
- unsigned long fibo (int n, unsigned long ∗vals)
- int **main** ()

Variables

— double t_values [ ]
Tableau contenant les quantiles pour le calcul de l'intervalle de confiance.

2.1.1 Description détaillée

Auteur

Jérémy ZANGLA (zangla.jeremy@gmail.com)

Version

0.1

Date

2019-11-01

Copyright

2.1.2 Documentation des fonctions

2.1.2.1 calc_radius()

```
double calc_radius (
            int nb,
            double * vals,
            double variance )
```

Calcule le rayon de l'intervalle de confiance.

Paramètres

nb	Nombre de valeur dans le jeu.
vals	Tableau contenant les valeurs du jeu.
variance	Variance du jeu de valeurs.

Renvoie

double Rayon de l'intervalle de confiance.

2.1.2.2 calc_variance()

```
double calc_variance (
            int nb,
            double * vals,
            double mean )
```

Calcule de la variance du jeu de valeur.

Paramètres

nb	Nombre de valeur dans le jeu.
vals	Tableau contenant les valeurs.
mean	Moyenne des valeurs du tableau.

Renvoie

double La variance du jeu.

2.1.2.3 fibo()

```
unsigned long fibo (
            int n,
            unsigned long * vals )
```

Paramètres

n	
vals	

Renvoie

unsigned long

2.1.2.4 get_t()

```
double get_t (
            int nb )
```

Permet d'obtenir le quantile correspondant au nombre de valeurs que l'on utilisent.

Les 30 premières valeurs du tableau sont en accès direct.
Les suivantes correspondent à des intervalles :
[30 ; 40[: case 30 du tableau
[40 ; 80[: case 31 du tableau
[80 ; 120[: case 32 du tableau
[120 ; +inf[: case 33 du tableau

Paramètres

nb	Nombre de valeurs utilisées.

Renvoie

double Valeur du quantile.

2.1.2.5 monte_carlo()

```
double monte_carlo (
            int nb_points )
```

Calcule la valeur de PI en utilisant la méthode de Monte-Carlo.

La méthode de Monte-Carlo utilise un cercle de rayon 1 centré en (0 ; 0) pour calculer PI. Cette méthode utilise la surface du cercle et celle du carrée pour calculer PI, respectivement PI∗r² = PI et c² = 4 où r est le rayon du cercle (1) et c est le coté du carré (2). Le rapport de ces surfaces donne la probabilité que le point généré soit dans le cercle. On peut donc déduire PI de la manière suivante :

$$\begin{aligned} p &= \frac{surface_{cercle}}{surface_{carre}} \\ \frac{nb\_inner}{nb\_points} &= \frac{surface_{cercle}}{surface_{carre}} \\ \frac{nb\_inner}{nb\_points} &= \frac{\pi \times r^2}{c^2} \\ \frac{nb\_inner}{nb\_points} &= \frac{\pi}{4} \\ 4 * \frac{nb\_inner}{nb\_points} &= \pi \end{aligned}$$

Paramètres

nbPoints	Nombre de points générés pour calculer PI.

Renvoie

double Valeur de PI calculée.

2.1.2.6 replicates_monte_carlo()

```
double replicates_monte_carlo (
            int nb_replicates,
            double * values )
```

Réitère la méthode de Monte-Carlo pour en faire des statistiques.

Chaque valeur de utilisée pour les statistiques sera calculée à parti de 1 000 000 000 de points.

Paramètres

nbReplicates	Nombre de valeur que l'on veut utilisé pour les futures statistiques.
values	Tableau dans lequel on va mettre les valeurs de calculées.

Renvoie

double Moyenne de toutes les valeurs calculées.

2.1.3 Documentation des variables

2.1.3.1 t_values

`t_values`

Valeur initiale :

```
= {
    12.706, 4.303, 3.182, 2.776, 2.571, 2.447, 2.365, 2.308, 2.262, 2.228, 2.201, 2.179, 2.160, 2.145, 2.13
      1, 2.120, 2.110, 2.101, 2.093, 2.086, 2.080, 2.074, 2.069, 2.064, 2.060, 2.056, 2.020, 2.048, 2.045, 2.042,
      2.021, 2, 1.980, 1.960
}
```

Tableau contenant les quantiles pour le calcul de l'intervalle de confiance.

Index

calc_radius
    main.c, 4
calc_variance
    main.c, 4

fibo
    main.c, 5

get_t
    main.c, 5

main.c, 3
    calc_radius, 4
    calc_variance, 4
    fibo, 5
    get_t, 5
    monte_carlo, 6
    replicates_monte_carlo, 6
    t_values, 7
monte_carlo
    main.c, 6

replicates_monte_carlo
    main.c, 6

t_values
    main.c, 7