

# Etape à suivre et pourquoi

---

## source\_pi\_direct.c

### Compilation

```
gcc -o exec_direct source_pi_direct.c mt.c -lm
```

### Execution

On remarque que le programme met 1.2 sec avec  $10 * 1\,000\,000$  de points

## source\_pi\_precompile.c

### Etape à suivre

#### Generer le tableau

```
# Compilation programme de génération du tableau
gcc -o generate_tab generate_tab.c mt.c -lm
# Execution du programme de génération du tableau
./generate_tab
# Compilation du tableau contenant les valeurs randoms
gcc -c tab.c -o tab.o
```

#### Compilation du programme

```
# Compilation du programme
gcc -c source_pi_precompile.c -o source_pi_precompile.o -lm

# Compilation du programme avec le tableau (lien)
gcc source_pi_precompile.o tab.o -o exec_precompile
```

### Execution

On remarque que le programme met 0,6 sec avec  $10 * 1\,000\,000$  de points

C'est formidable, on a gagné 0,6 sec soit la moitié du temps d'execution