

## REPORTE TAREA #1

### 1. No protejan la variable `points_inside_circle` con un mutex.

1 Hilo:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1"
# gcc -pthread -o executeme calcPi_pthread01.c -lm
time ./executeme 1000000000 1

Valor de pi: 3.141594

272.97user 0.00system 4:32.98elapsed 99%CPU (0avgtext+0avgdata 11048maxresident)k
0inputs+1024outputs (0major+2422minor)pagefaults 0swaps
```

4 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 4"
# gcc -pthread -o executeme calcPi_pthread01.c -lm
time ./executeme 1000000000 4

Valor de pi: 0.785386

59.55user 0.00system 0:59.55elapsed 99%CPU (0avgtext+0avgdata 10992maxresident)k
0inputs+1024outputs (0major+2419minor)pagefaults 0swaps
```

8 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8"
# gcc -pthread -o executeme calcPi_pthread01.c -lm
time ./executeme 1000000000 8

Valor de pi: 0.392699

32.75user 0.00system 0:32.76elapsed 99%CPU (0avgtext+0avgdata 10968maxresident)k
0inputs+1024outputs (0major+2422minor)pagefaults 0swaps
```

Mientras más hilos sean añadidos, mas impreciso es el resultado, pero más rápido sale.0 errores con valgrind

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8" valgrind
gcc -pthread -g -o executeme calcPi_pthread01.c -lm
valgrind --tool=helgrind time ./executeme 1000000000 8
==33739== Helgrind, a thread error detector
==33739== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==33739== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==33739== Command: time ./executeme 1000000000 8
==33739==

Valor de pi: 0.454014

55.39user 0.00system 0:07.76elapsed 713%CPU (0avgtext+0avgdata 37044maxresident)k
0inputs+0outputs (0major+966minor)pagefaults 0swaps
==33739==
==33739== Use --history-level=approx or =none to gain increased speed, at
==33739== the cost of reduced accuracy of conflicting-access information
==33739== For lists of detected and suppressed errors, rerun with: -s
==33739== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

## 2. Protejan la variable `points_inside_circle` con un mutex dentro del ciclo for.

1 Hilo:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1"
# gcc -pthread -o executeme calcPi_pthread02.c -lm
time ./executeme 1000000000 1

Valor de pi: 3.141586

242.36user 0.04system 4:02.41elapsed 99%CPU (0avgtext+0avgdata 11032maxresident)k
0inputs+1024outputs (0major+2419minor)pagefaults 0swaps
```

4 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 4"
# gcc -pthread -o executeme calcPi_pthread02.c -lm
time ./executeme 1000000000 4

Valor de pi: 0.785400

59.78user 0.00system 0:59.79elapsed 99%CPU (0avgtext+0avgdata 10972maxresident)k
0inputs+1024outputs (0major+2424minor)pagefaults 0swaps
```

8 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8"
# gcc -pthread -o executeme calcPi_pthread02.c -lm
time ./executeme 1000000000 8

Valor de pi: 0.392696

29.98user 0.00system 0:29.99elapsed 99%CPU (0avgtext+0avgdata 11032maxresident)k
0inputs+1024outputs (0major+2420minor)pagefaults 0swaps
```

Lo mismo que el punto #1, al correrlo con Valgrind 0 errores:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8" valgrind
gcc -pthread -g -o executeme calcPi_pthread02.c -lm
valgrind --tool=helgrind time ./executeme 1000000000 8
==33789== Helgrind, a thread error detector
==33789== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==33789== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==33789== Command: time ./executeme 1000000000 8
==33789==

Valor de pi: 3.141624

97.93user 283.12system 0:53.01elapsed 718%CPU (0avgtext+0avgdata 37044maxresident)k
0inputs+0outputs (0major+869minor)pagefaults 0swaps
==33789==
==33789== Use --history-level=approx or =none to gain increased speed, at
==33789== the cost of reduced accuracy of conflicting-access information
==33789== For lists of detected and suppressed errors, rerun with: -s
==33789== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

### 3. Protejan la variable `points_inside_circle` con un mutex fuera del ciclo `for`. En lugar de incrementar la variable global `points_inside_circle` dentro del ciclo, actualicen una variable local y al final del ciclo incrementen la variable global.

1 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1"
# gcc -pthread -o executeme calcPi_pthread03.c -lm
time ./executeme 1000000000 1

Valor de pi: 3.141594

17.52user 0.00system 0:17.52elapsed 99%CPU (0avgtext+0avgdata 2048maxresident)k
0inputs+0outputs (0major+87minor)pagefaults 0swaps
```

4 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 4"
# gcc -pthread -o executeme calcPi_pthread03.c -lm
time ./executeme 1000000000 4

Valor de pi: 3.141543

74.98user 91.01system 0:44.21elapsed 375%CPU (0avgtext+0avgdata 2188maxresident)k
0inputs+0outputs (0major+94minor)pagefaults 0swaps
```

8 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8"
# gcc -pthread -o executeme calcPi_pthread03.c -lm
time ./executeme 1000000000 8

Valor de pi: 3.141598

99.96user 285.32system 0:53.54elapsed 719%CPU (0avgtext+0avgdata 2252maxresident)k
0inputs+0outputs (0major+101minor)pagefaults 0swaps
```

En este caso mientras más hilos usemos, más tiempo conlleva realizar el cálculo, solo que la precisión en este caso es mucho mejor en cualquier escenario que los otros códigos. 0 errores en valgrind

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1" valgrind
gcc -pthread -g -o executeme calcPi_pthread03.c -lm
valgrind --tool=helgrind time ./executeme 1000000000 1
==34089== Helgrind, a thread error detector
==34089== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==34089== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==34089== Command: time ./executeme 1000000000 1
==34089==

Valor de pi: 3.141583

13.58user 0.00system 0:13.58elapsed 99%CPU (0avgtext+0avgdata 37044maxresident)k
0inputs+0outputs (0major+953minor)pagefaults 0swaps
==34089==
==34089== Use --history-level=approx or =none to gain increased speed, at
==34089== the cost of reduced accuracy of conflicting-access information
==34089== For lists of detected and suppressed errors, rerun with: -s
==34089== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

**4. Similar a 3, pero no actualicen `points_inside_circle` dentro de `throw_darts`. Retornen la cantidad de puntos dentro del círculo (usando un falso puntero) y hagan que el hilo principal incremente la variable global.**

1 Hilo:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1"
# gcc -pthread -o executeme calcPi_pthread04.c -lm
time ./executeme 1000000000 1

Valor de pi: 3.141585

13.56user 0.00system 0:13.56elapsed 99%CPU (0avgtext+0avgdata 2656maxresident)k
0inputs+0outputs (0major+185minor)pagefaults 0swaps
```

4 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 4"
# gcc -pthread -o executeme calcPi_pthread04.c -lm
time ./executeme 1000000000 4

Valor de pi: 3.141619

13.68user 0.00system 0:03.43elapsed 398%CPU (0avgtext+0avgdata 2624maxresident)k
0inputs+0outputs (0major+193minor)pagefaults 0swaps
```

8 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8"
# gcc -pthread -o executeme calcPi_pthread04.c -lm
time ./executeme 1000000000 8

Valor de pi: 3.141611

15.55user 0.00system 0:02.13elapsed 728%CPU (0avgtext+0avgdata 2636maxresident)k
0inputs+0outputs (0major+199minor)pagefaults 0swaps
```

En este caso mientras más hilos usemos, más rápido hace el cálculo, la precisión es bastante buena. 0 errores en valgrind

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8" valgrind
gcc -pthread -g -o executeme calcPi_pthread04.c -lm
valgrind --tool=helgrind time ./executeme 1000000000 8
==34511== Helgrind, a thread error detector
==34511== Copyright (C) 2007-2017, and GNU GPL'd, by OpenWorks LLP et al.
==34511== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==34511== Command: time ./executeme 1000000000 8
==34511==

Valor de pi: 3.141513

15.79user 0.00system 0:02.47elapsed 638%CPU (0avgtext+0avgdata 37044maxresident)k
0inputs+0outputs (0major+967minor)pagefaults 0swaps
==34511==
==34511== Use --history-level=approx or =none to gain increased speed, at
==34511== the cost of reduced accuracy of conflicting-access information
==34511== For lists of detected and suppressed errors, rerun with: -s
==34511== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

## 5. Usen operaciones atómicas para incrementar la variable `points_inside_circle`.

1 Hilo:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 1"
gcc -pthread -o executeme calcPi_pthread05.c -lm
time ./executeme 1000000000 1

Valor de pi: 3.141582

13.59user 0.00system 0:13.59elapsed 99%CPU (0avgtext+0avgdata 2616maxresident)k
0inputs+0outputs (0major+184minor)pagefaults 0swaps
```

4 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 4"
gcc -pthread -o executeme calcPi_pthread05.c -lm
time ./executeme 1000000000 4

Valor de pi: 3.141618

14.41user 0.00system 0:04.17elapsed 344%CPU (0avgtext+0avgdata 2716maxresident)k
0inputs+0outputs (0major+192minor)pagefaults 0swaps
```

8 Hilos:

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.Tarea1$ make ARGS="1000000000 8"
gcc -pthread -o executeme calcPi_pthread05.c -lm
time ./executeme 1000000000 8

Valor de pi: 3.141631

15.73user 0.00system 0:02.45elapsed 639%CPU (0avgtext+0avgdata 2548maxresident)k
0inputs+0outputs (0major+199minor)pagefaults 0swaps
```

Conclusión:

Las implementaciones más rápidas eran en las que el mutex intervenía solo para modificar un único dato por hilo y no quedarse dentro de un for modificando datos constantemente.