

Raport

Zarządzanie pamięcią, biblioteki, pomiar czasu

Zadanie 1. Kompilator, optymalizacja, pomiar czasu

Programy można skompilować za pomocą „make test”, a uruchomić „./run-all.sh”

1. fibonacci.c – Ciąg Fibonacciego iteracyjnie

Flaga	Czas wykonania (s)
(brak)	4.407471
-O1	0.545492
-O2	0.000001
-O3	0.000001

Widzimy znaczne różnice w czasie wykonania - kompilator od -O2 w górę całkowicie pomija obliczenia, prawdopodobnie z racji tego że wynik nie jest nigdzie wykorzystywany), co można sprawdzić w disassembly (cała główna pętla programu pominięta).

2. sito.c – Sito Erastotenesa

Flaga	Czas wykonania (s)
(brak)	1.294993
-O1	1.162374
-O2	1.150555
-O3	1.179913

Zauważamy niewielki spadek od flagi -O1, później nie widać wielkiej różnicy.

3. fibonacci_rec.c – Ciąg fibonacciego rekurencyjnie

Flaga	Czas wykonania (s)
(brak)	segfault
-O1	0.004316
-O2	0.000652
-O3	0.000592

Program napisano z rekurencją ogonową. Dla $N=2e5$ program bez optymalizacji kończy się segfaultem. W kolejnych krokach widać regularne spadki.

4. bubblesort.c – Sortowanie bąbelkowe

Flaga	Czas wykonania (s)
(brak)	42.513327
-O1	8.118990
-O2	19.155112
-O3	11.287449

Zauważamy wyraźny spadek przy -O1, później czas natomiast rośnie – możliwe ograniczenia termiczne.