

Mateusz Obara, 264092

Pracownia 3: „Optymalizacja Kodu”

Informacje o systemie

Dystrybucja: Fedora 29 64bit

Jądro systemu: 5.0.7-200

Kompilator: gcc version 8.3.1

Procesor: Intel(R) Core(TM) i5 CPU M 520 @ 2.40GHz

Liczba rdzeni: 2, z hyperthreadingiem

Pamięć podręczna

L1d: 32 KiB, 8-drożny (per rdzeń), rozmiar linii 64 B

L2: 256 KiB, 8-drożny (per rdzeń), rozmiar linii 64 B

L3: 3 MiB, 12-drożny (współdzielony), rozmiar linii 64 B

Pamięć TLB

L1d: 4 KiB strony, 4-drożny, 64 wpisy

L2: 4 KiB strony, 4-drożny, 512 wpisów

Informacje o pamięciach podręcznych uzyskano na podstawie wydruku programów getconf, x86info oraz cpuid.

Generowanie wyników

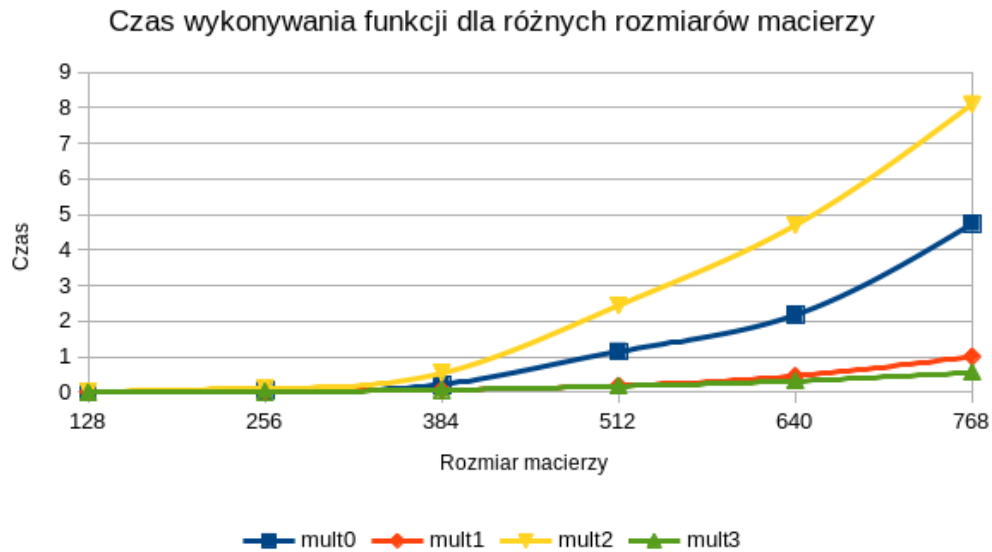
Skrypty rmaker1-1.sh, rmaker1-2.sh, rmaker3-1.sh, rmaker3-2.sh, rmaker4.sh służą do powtórzenia eksperymentu dotyczącego odpowiadającego im zadania i pytania (np rmaker3-2.sh generuje wyniki koniecznie do powiedzi na 2 pytanie w zadaniu 3). Można je również wywołać wszystkie po kolei, za pomocą rmakerall.sh.

Generowanie wyników może trochę zająć, dlatego dołączam również ostatnie wyniki utworzone na mojej maszynie.

Zadanie 1

Czy wyniki różnią się od tych uzyskanych na slajdzie?

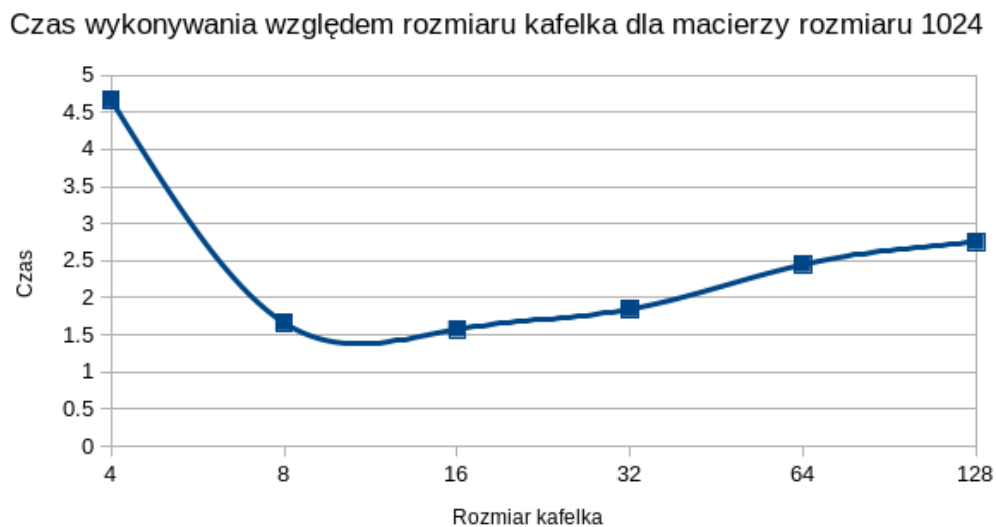
Tak, różnią się, ale wykazują wspólny trend. Wyniki uzyskane na mojej maszynie wydają się bardziej 'wachać'.



Z czego wynika rozbieżność między wynikami dla poszczególnych wersji mnożenia macierzy?

Wynika ona z różnej kolejności pobierania danych z pamięci. Szybsze funkcje starają się czytać je tak, aby ograniczyć liczbę missów w cacheu.

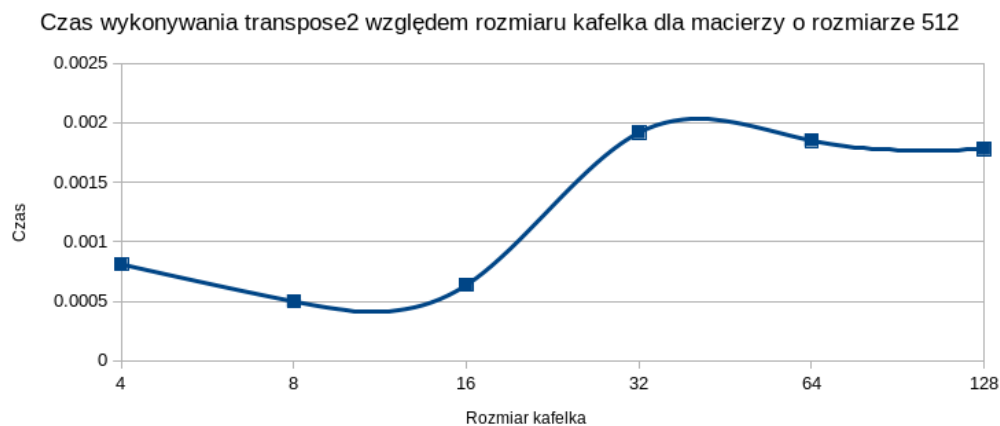
Jaki wpływ ma rozmiar kafelka na wydajność multiply?



Z danych możemy wyczytać że funkcja działa najszybciej gdy rozmiar kafelka wynosi 16. Kafelki rozmiaru 8 również prezentuje dobre wyniki. Dla reszty, zarówno dla mniejszych i większych wartości obserwujemy dłuższe czasy.

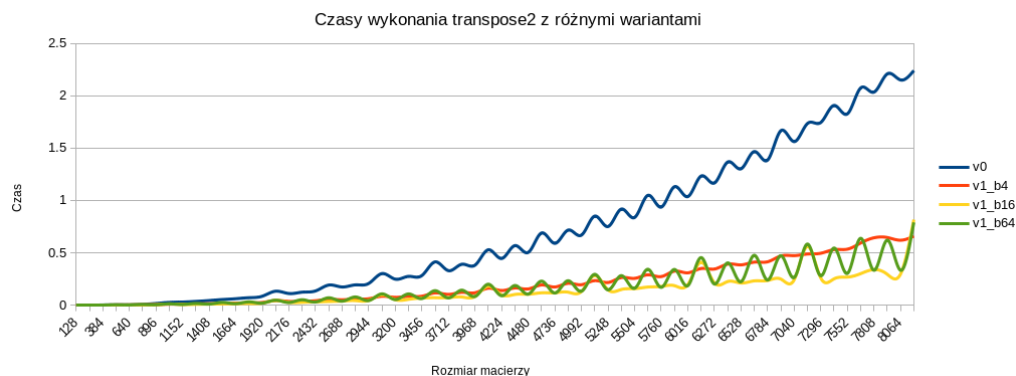
Zadanie 3

Jaki wpływ na wydajność transpose2 ma rozmiar kafelka?



Z zebranych danych wynika że transpose2 jest najwydajniejsza przy kafelku o rozmiarze 8.

Czy czas wykonania programu z różnymi rozmiarami macierzy identyfikuje rozmiary poszczególnych poziomów pamięci podręcznej?



Z wykresu możemy wyczytać cykliczne wzrosty w czasie wykonywania programu. Zauważmy też że wersja bezkafelkowa takich nie posiada. Można zatem wywnioskować że zależne są od kafelków - czyli mniejszych wersji macierzy. O ile można by oszacować rozmiary dla L1 i L2, L3 jest współdzielona i pozostaje problemem.

Zadanie 4

Ile instrukcji maszynowych ma ciało pętli przed i po optymalizacji?

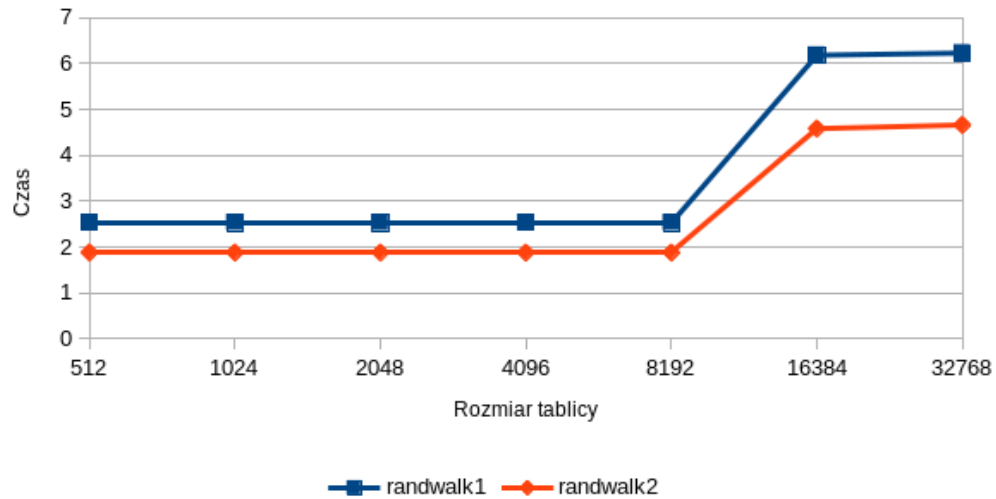
Przed optymalizacją: 57 Po optymalizacji: 58

Ile spośród nich to instrukcje warunkowe?

Przed optymalizacją: 6 Po optymalizacji: 3

Czy rozmiar tablicy ma duży wpływ na działanie programu?

Czas wykonywania różnych wariantów randwalk względem wielkości tablicy



Tak. podążając za wykresem, widzimy że na początku, dla dużej ilości danych czas wykonywania funkcji nie zmienia się znacznie. Jednak, na rozmiarze 16384 zauważamy znaczny skok, a następnie znów niewielką zmianę. Wspomniany wzrost sugeruje że czas wykonywania randwalk jest jednak zależny od rozmiaru tablicy.