

Systemy operacyjne 2		
Informatyka S1	Semestr 4	2019/2020
Laboratorium 4		

Wstęp teoretyczny:

- Obsługa plików z wykorzystaniem f-cji bibliotecznych (fopen, getline), rozdzielanie łańcuchów znaków na podłańcuchy/tokeny (strtok).
- Obsługa wątków PTHREAD (f-cje pthread\_...).
- Precyzyjne pomiary czasu (f-cje clock\_gettime, pthread\_getcpuclockid).

Zadanie:

- Napisać program, który będzie pobierał dwa argumenty wywołania. Pierwszy argument będzie interpretowany jako nazwa pliku zawierającego macierz, drugi to ilość wątków, które będą wykonywały obliczenia. Zakładamy, że plik z macierzą jest plikiem tekstowym, elementy macierzy to tekstowa reprezentacja liczb od 0 do 255, a ilość kolumn i wierszy w macierzy jest parzysta (nie musimy tego sprawdzać). Przykładowy plik o nazwie **matrix\_1000\_2000.txt** znajduje się na stronie z materiałami. Liczba wątków, jaką można podać jako argument, to **1, 2, 4** lub **8**. Zadaniem programu jest przeskalowanie macierzy w taki sposób, aby ilość kolumn i wierszy zmniejszyła się dwukrotnie, przy czym nowe elementy liczone są jako wartość średnia czterech sąsiednich elementów. Przykładowo:

0	120	105	34	81	210	->	34	42	76
10	6	13	7	12	1				
20	10	25	43	18	10		45	23	58
140	10	15	7	102	101				

- Obliczenia powinny zostać rozdzielone pomiędzy określoną w wywołaniu liczbę wątków.
- Wynikowa macierz powinna zostać zapisana do nowoutworzonego pliku.
- Program powinien wyświetlić czas jaki był potrzebny na przeprowadzenie obliczeń.
- Sugerowane jest odczytanie zawartości pliku przed rozpoczęciem działania poszczególnych wątków.

Uwaga! Kod źródłowy programu (1 plik) po oddaniu prowadzącemu zajęcia laboratoryjne musi zostać przesłany na adres [so2@zut.edu.pl](mailto:so2@zut.edu.pl):

- plik z kodem źródłowym musi mieć nazwę **lab04.main.c**
- do maila musi być dołączony plik **manifest.json**. Plik manifestu dla konkretnego zadania można pobrać ze strony z materiałami. W pliku manifestu należy wpisać poprawne wartości **email**, **imię**,

**nazwisko, indeks i grupa.** W polach **kompilacja** i **uruchomienie** wpisujemy polecenia, za pomocą których dokonujemy kompilacji i testowego uruchomienia. W wszystkich polach są wprowadzone przykładowe dane,

- mail musi zostać wysłane z poczty uczelnianej (**zut.edu.pl**),
- nagłówek maila powinien zawierać przynajmniej informację o numerze zadania, np.: **SO2 S1 LAB04**

Dostarczone kody źródłowe będą analizowane pod kątem występowania plagiatów, kompilowane i uruchamiane (zgodnie z informacją przesłaną w manifeście). Niewysłanie wiadomości, wysłanie jej w formie niezgodnej z powyższymi wymaganiami lub wysłanie kodu, który się nie kompiluje i nie uruchamia będzie traktowane jako brak programu i skutkowało otrzymaniem za niego oceny niedostatecznej.