

Zasady pomiaru czasu wykonania programów i ich wybranych fragmentów.

1. Do pomiaru czasu wykonania całego programu można wykorzystać narzędzie systemowe *time* lub bardziej zaawansowane narzędzia analizy wydajności: profilery, takie jak *gprof*, interfejsy z licznikami sprzętowymi, takie jak *perf*, i inne
2. Do pomiaru czasu wykonania fragmentów programów można zastosować wywołania funkcji systemowych zwracających wykorzystanie zasobów systemu, np. czasu CPU, takich jak *getrusage()*, lub odmierzających czas zewnętrzny, takich jak *gettimeofday()* (ewentualnie funkcji udostępnianych w ramach laboratorium stanowiących prosty interfejs do funkcji systemowych)
-> uwaga: w przypadku programów wielowątkowych i ich fragmentów czas CPU jest sumą czasów CPU wszystkich wątków (porównanie czasu CPU i zewnętrznego (zegarowego) może służyć za wskazówkę dotyczącą wielowątkowości wykonania)
3. Do pomiaru czasu zewnętrznego (*wall clock time*) można także użyć funkcji dostarczanych przez środowiska tworzenia programów równoległych – *omp_get_wtime()* - OpenMP, *MPI_Wtime()* - MPI
4. Każdy pomiar czasu obarczony jest mniejszym lub większym błędem (zakłócenia systemowe, niedokładności odczytu zegarów i liczników, itp.)
5. **W celu eliminacji losowych zaburzeń pomiaru czasu należy:**
 1. **powtórzyć pomiar kilka razy (minimum 3)**
 2. **odrzuć wyniki zdecydowanie odbiegające od pozostałych (przy znacznym rozrzucie wyników może być konieczne wykonanie więcej niż 3 pomiarów)**
 3. **wyciągnąć wartość średnią z pomiarów wiarygodnych (inną możliwością jest przyjęcie wartości minimalnej z kilku pomiarów – zaburzenia zazwyczaj zwiększają czas wykonania kodu)**