

# PR

## ”OpenMP - podstawy”

### Opracowanie zadania 2

Mateusz Chodyła

19 kwietnia 2020

## 1 Opis środowiska

- OS : Linux (Ubuntu 18.04) x64
- Procesor : Intel Core i5 4460 3.2GHz (4 rdzenie fizyczne, 4 rdzenie logiczne)

## 2 Kod programu

---

```
#include <stdio.h>
#include <time.h>
#include <omp.h>

double step, x, pi, sum=0.0, start_omp, stop_omp;
int i;
clock_t start, stop;

int pi_1(long long num_steps)
{
    step = 1./((double)num_steps;
    start = clock();           // Systemowy pomiar czasu
    start_omp = omp_get_wtime();// Pomiar czasu rzeczywistego (funkcja OMP)

    for (i=0; i<num_steps; i++)
    {
        x = (i + .5)*step;
        sum = sum + 4.0/(1.+ x*x);
    }

    pi = sum*step;
    stop = clock();
    stop_omp = omp_get_wtime();

    printf("Wartosc liczby PI wynosi %15.12f\n",pi);
    printf("Czas przetwarzania wynosi:\n\tSYS : %f sekund\n\tOMP : %f\n\tsekund",((double)(stop - start)/ CLOCKS_PER_SEC), stop_omp-start_omp);
```

```
    return 1;
}
int pi_2(long long num_steps, int num_threads){
    return 2;
}
int pi_3(long long num_steps){
    return 3;
}
int pi_4(long long num_steps){
    return 4;
}
int pi_5(long long num_steps){
    return 5;
}
int pi_6(long long num_steps){
    return 6;
}
int pi_7(long long num_steps){
    return 7;
}
```

---