|  |  |
| --- | --- |
| Akademia Nauk Stosowanych w Nowym Sączu  Wydział Nauk Inżynieryjnych | |
| Imię i nazwisko: | Michał Bernardy |
| Grupa: | P1 |
| Ocena: |  |

**Wstęp**

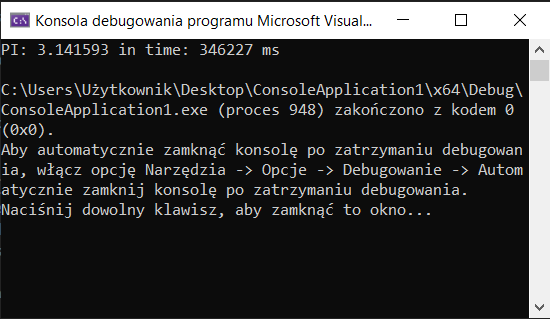
Celem niniejszego sprawozdania jest analiza działania programu obliczającego wartość liczby π metodą numeryczną z wykorzystaniem wielowątkowości w języku C++. Obliczenia realizowane są równolegle przy użyciu biblioteki std::thread, gdzie każdy wątek wykonuje część sumowania, a następnie wyniki są sumowane w fazie redukcji.

W ramach zadania sprawdzono, jak zmiana liczby kroków (steps) oraz liczby wątków (threads\_count) wpływa na dokładność wyniku oraz czas wykonania programu. Celem eksperymentów było porównanie efektywności obliczeń w różnych konfiguracjach oraz ocena wpływu równoległości na wydajność.

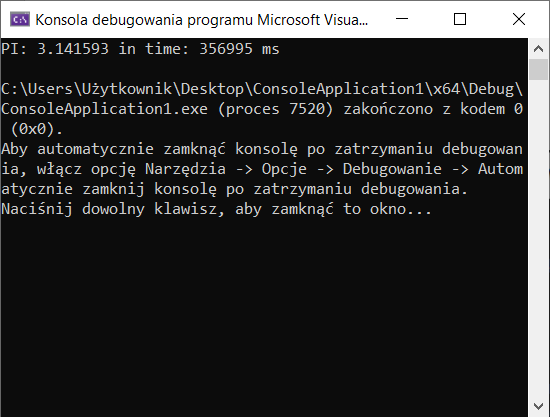
**Wyniki**

**KOD5**

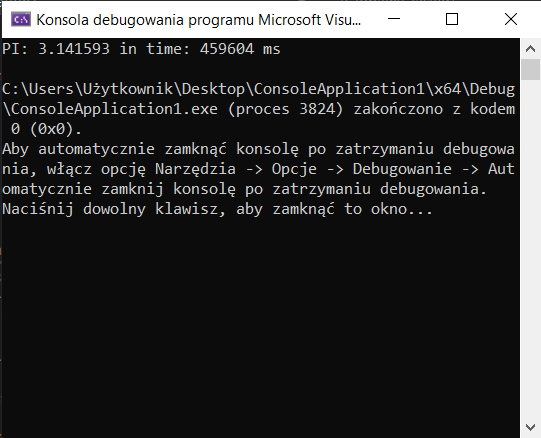
10 steps: 90000000000



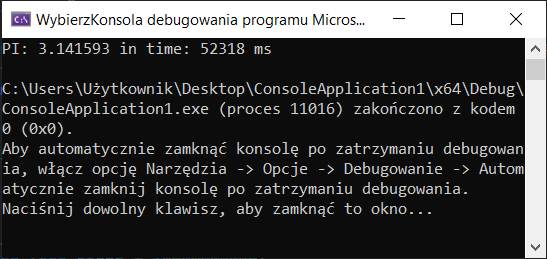
5 steps: 90000000000



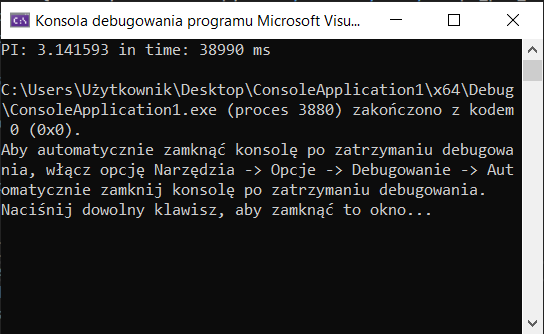
3 steps: 90000000000



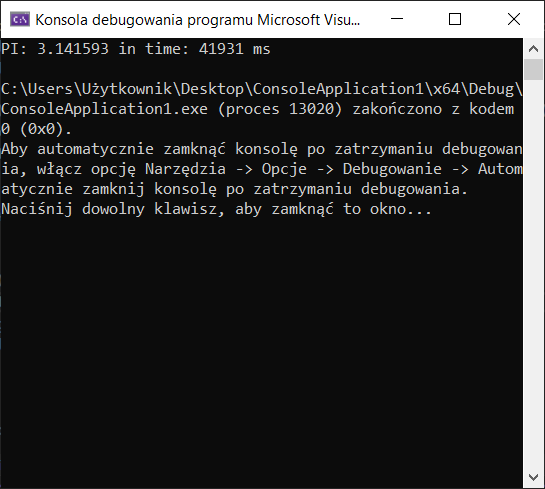
3 steps: 10000000000



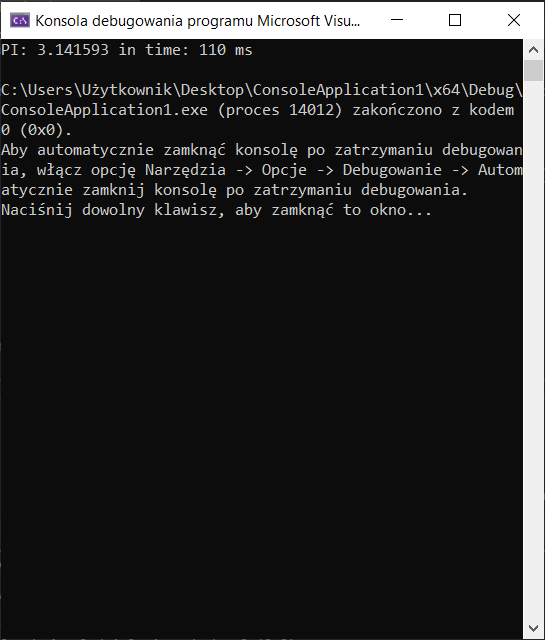
5 steps: 10000000000



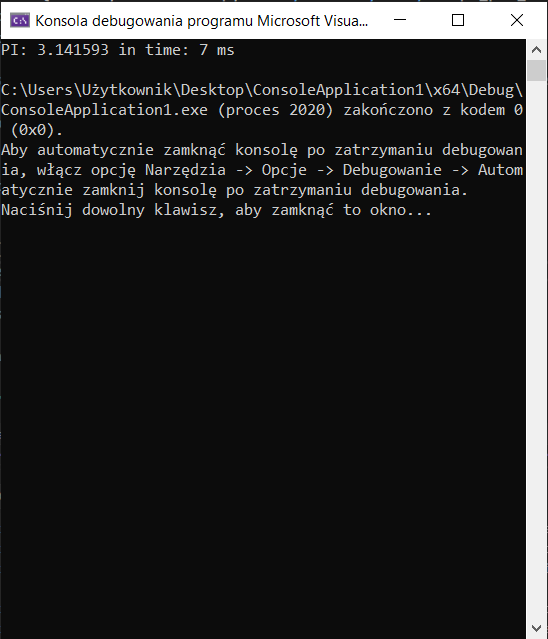
10 steps: 10000000000



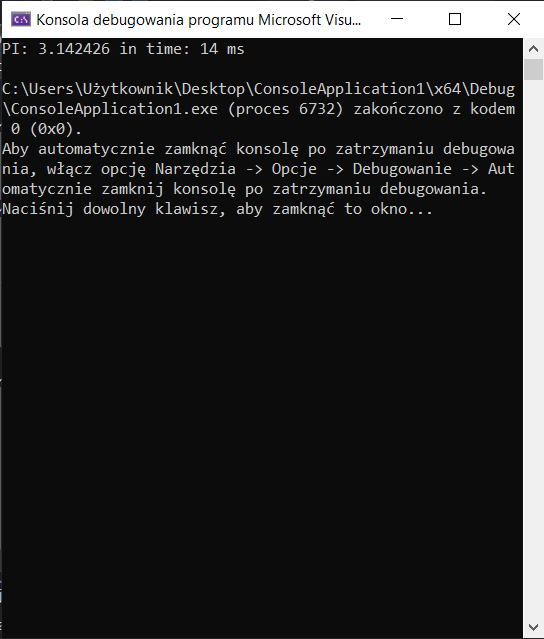
10 steps: 1000



1 steps: 1000

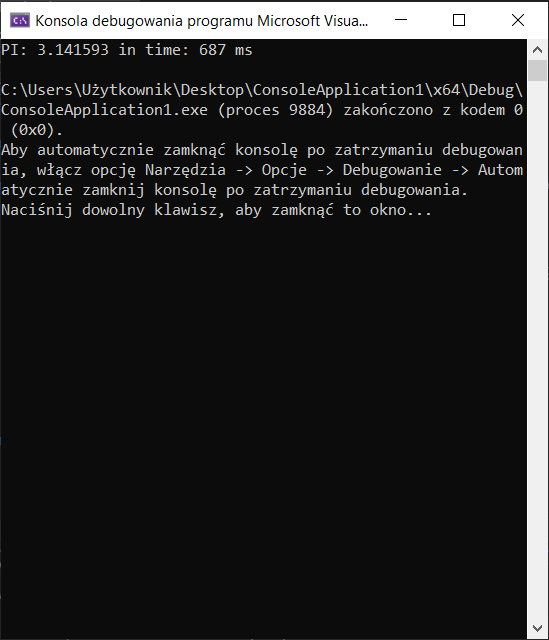


1 steps: 10

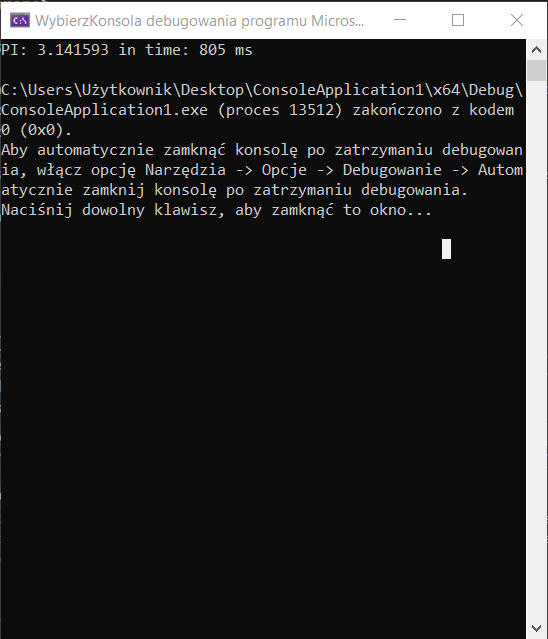


**KOD6:**

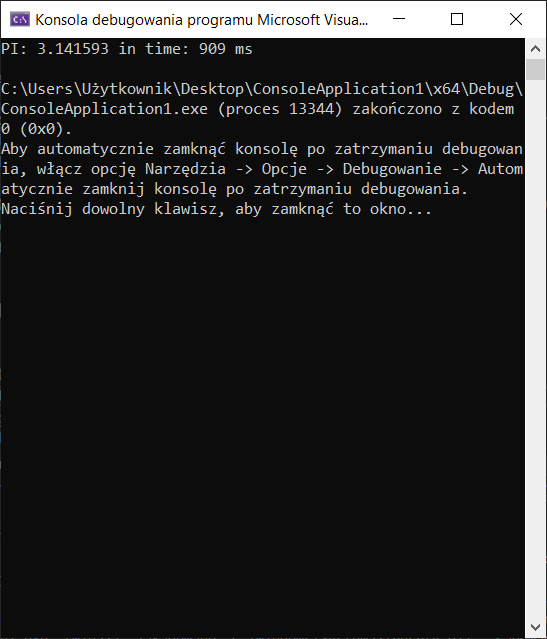
10 steps: 100000000



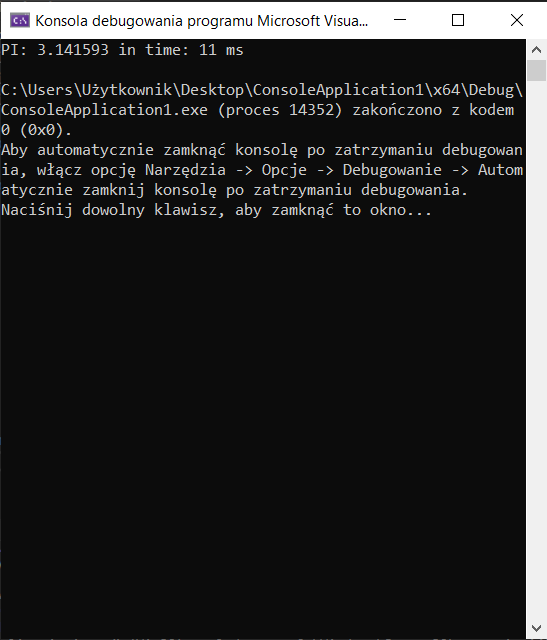
5 steps: 100000000



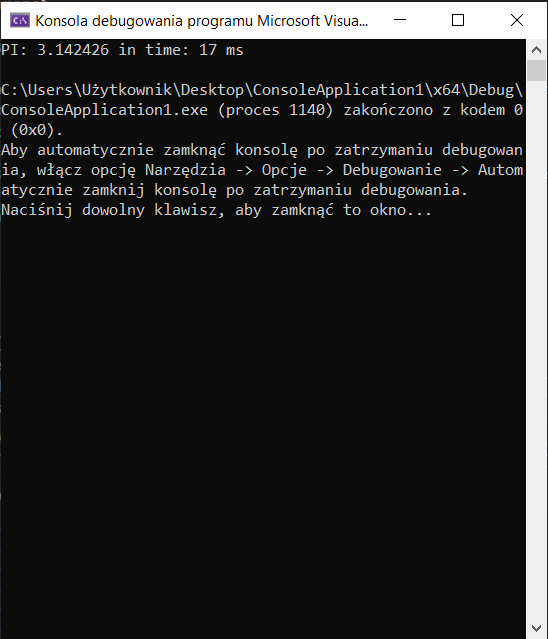
3 steps: 100000000



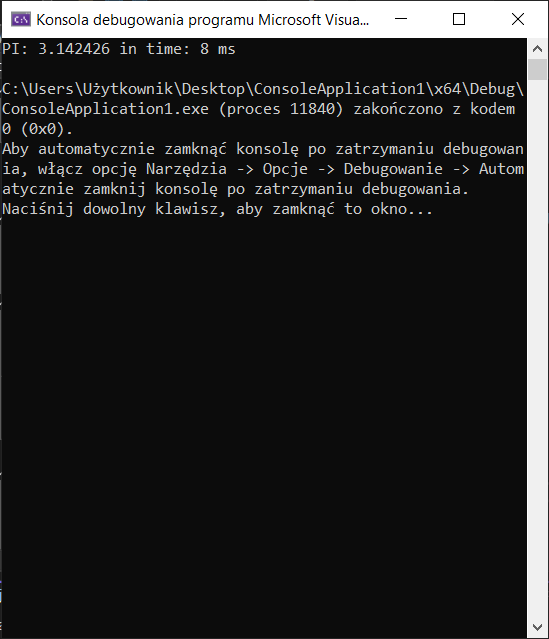
1 steps: 1000



10 steps: 10



1 steps:10



**Wnioski**

Na podstawie przeprowadzonych testów można stwierdzić, że dokładność obliczeń liczby π rośnie wraz ze zwiększeniem liczby kroków (steps). Przy małej liczbie kroków (np. 10) wyniki są niedokładne i odbiegają od rzeczywistej wartości liczby π. Z kolei przy dużej liczbie kroków wynik jest znacznie bardziej precyzyjny.

Jednocześnie zauważono, że czas wykonania programu rośnie wraz ze zwiększaniem wartości steps, co wynika z większej liczby operacji do wykonania przez procesor. Wydłużony czas jest więc naturalną konsekwencją dążenia do większej dokładności oblicze