

proj5

Generated by Doxygen 1.12.0

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 src/main.cpp File Reference	3
2.1.1 Function Documentation	3
2.1.1.1 generateBenchmarks()	3
2.1.1.2 getPi()	4
2.1.1.3 main()	4
Index	5

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

src/ main.cpp	3
---	---

Chapter 2

File Documentation

2.1 src/main.cpp File Reference

```
#include <cassert>
#include <cmath>
#include <iostream>
#include <thread>
#include <print>
#include <vector>
#include <fstream>
```

Functions

- double [getPi](#) (const size_t steps, const size_t threads)
Oblicza wartość liczby Pi metodą przybliżonej całki używając określonej liczby wątków i dokładności.
- void [generateBenchmarks](#) (const size_t minThreads, const size_t maxThreads, const size_t minIter, const size_t maxIter)
Generuje pliki .csv benchmarków dla obliczania liczby Pi z różną liczbą wątków i iteracji.
- int [main](#) (int argc, char *argv[])

2.1.1 Function Documentation

2.1.1.1 generateBenchmarks()

```
void generateBenchmarks (
    const size_t minThreads,
    const size_t maxThreads,
    const size_t minIter,
    const size_t maxIter)
```

Generuje pliki .csv benchmarków dla obliczania liczby Pi z różną liczbą wątków i iteracji.

Parameters

<i>minThreads</i>	Minimalna liczba wątków do użycia w benchmarkach.
<i>maxThreads</i>	Maksymalna liczba wątków do użycia w benchmarkach.
<i>minIter</i>	Minimalna liczba iteracji do użycia w benchmarkach.
<i>maxIter</i>	Maksymalna liczba iteracji do użycia w benchmarkach.

Note

Benchmarki są zapisywane do pliku `benchmark.csv`.

2.1.1.2 getPi()

```
double getPi (  
    const size_t steps,  
    const size_t threads)
```

Oblicza wartość liczby Pi metodą przybliżonej całki używając określonej liczby wątków i dokładności.

Parameters

<i>steps</i>	Liczba kroków do użycia w obliczeniach, określa dokładność.
<i>threads</i>	Liczba wątków do użycia w obliczeniach.

Returns

Obliczona wartość liczby Pi lub -1 w przypadku błędu.

2.1.1.3 main()

```
int main (  
    int argc,  
    char * argv[])
```


Index

generateBenchmarks

main.cpp, [3](#)

getPi

main.cpp, [4](#)

main

main.cpp, [4](#)

main.cpp

generateBenchmarks, [3](#)

getPi, [4](#)

main, [4](#)

src/main.cpp, [3](#)