

Program sluzacy do pomiaru zlozonosci obliczeniowej - Laboratorium 1

Wygenerowano przez Doxygen 1.8.9.1

Śr, 18 mar 2015 12:42:11

Spis treści

1	Program sluzacy do pomiaru zlozonosci obliczeniowej.	1
1.1	wartosci	2
2	Indeks klas	2
2.1	Lista klas	2
3	Indeks plików	2
3.1	Lista plików	2
4	Dokumentacja klas	2
4.1	Dokumentacja klasy dane	2
4.1.1	Opis szczegółowy	3
4.1.2	Dokumentacja konstruktora i destruktora	3
4.1.3	Dokumentacja funkcji składowych	3
4.1.4	Dokumentacja atrybutów składowych	5
5	Dokumentacja plików	5
5.1	Dokumentacja pliku generuj.cpp	5
5.1.1	Dokumentacja funkcji	5
5.2	Dokumentacja pliku generuj.hh	6
5.2.1	Dokumentacja funkcji	6
5.3	Dokumentacja pliku main.cpp	6
5.3.1	Dokumentacja funkcji	7
6	Zadanie	8
7	Realizacja	8
8	Działanie	8
9	Wyniki	8
10	Komentarz	8

1 Program sluzacy do pomiaru zlozonosci obliczeniowej.

Autor

Wojcich Makuch

Data

10.03.2015

Wersja

1.0

Zadaniem programu jest wygenerowanie tablic liczb psedoloswych oraz pomiar zlozonosci obliczeniowej polegajacej na wymnozeniu kazdego z tych elementow przez 2. Program zapisuje dane w pliku o nazwie Pomiar_czasu1.txt.

1.1 wartosci

Program wykonuje obliczenia dla tablic o rozmiarach: 10-10000000

2 Indeks klas

2.1 Lista klas

Tutaj znajdują się klasy, struktury, unie i interfejsy wraz z ich krótkimi opisami:

dane	2
----------------------	---

3 Indeks plików

3.1 Lista plików

Tutaj znajduje się lista wszystkich plików z ich krótkimi opisami:

generuj.cpp	5
generuj.hh	6
main.cpp	6

4 Dokumentacja klas

4.1 Dokumentacja klasy dane

```
#include <generuj.hh>
```

Metody publiczne

- [dane](#) (long int wielkosc)
- [~dane](#) ()
- void [generuj](#) ()
- int [Wez](#) (int i) const
- int [WezRozmiar](#) () const
- int [ZapiszDoPliku](#) (const char *nazwa)
- double [licz](#) ()

Atrybuty prywatne

- long int * [tablica](#)
- long int [rozmiar](#)

4.1.1 Opis szczegółowy

Klasa modeluje pojecie zbioru danych. Jej artybutem jest generowanie liczb pseldolosowych z zakresu 0-9 oraz pomiar zlozonosci obliczeniowej bazujacej na tych liczbach.

Definicja w linii 23 pliku generuj.hh.

4.1.2 Dokumentacja konstruktora i destruktor

4.1.2.1 `dane::dane (long int wielkosc) [inline]`

Parametry

<code>in</code>	<code>wielkosc</code>	- liczba elementow tablicy, dla ktorej zostanie przydzielona pamiec.
-----------------	-----------------------	----------------------------------------------------------------------

Definicja w linii 33 pliku generuj.hh.

4.1.2.2 `dane::~dane () [inline]`

Definicja w linii 34 pliku generuj.hh.

4.1.3 Dokumentacja funkcji skladowych

4.1.3.1 `void dane::generuj ()`

wypelnia tablice liczbami pseldolosowymi z zakresu 0-9.

Definicja w linii 23 pliku generuj.cpp.

4.1.3.2 `double dane::licz ()`

Wykonuje operacje na tablicy polegajace na wymnozeniu kazdego elementu przez 2. Mierzy czas wykonywanych operacji z dokladnoscia do e-06s.

Zwraca

tm - zmierzony czas.

Definicja w linii 31 pliku generuj.cpp.

4.1.3.3 `int dane::Wez (int i) const [inline]`

Parametry

<code>in</code>	<code>i</code>	- indeks tablicy.
-----------------	----------------	-------------------

Zwraca

zwraca element tablicy.

Definicja w linii 46 pliku generuj.hh.

4.1.3.4 `int dane::WezRozmiar () const [inline]`

Zwraca

Zwraca rozmiar tablicy.

Definicja w linii 51 pliku generuj.hh.

4.1.3.5 int dane::ZapiszDoPliku (const char * *nazwa*)

Zapisuje elementy tablicy w pliku o zadanej nazwie.

Parametry

<code>in</code>	<code>nazwa</code>	- nazwa pliku do zapisu.
-----------------	--------------------	--------------------------

Zwraca

0 - Poprawny zapis. -1 - Niepoprawny zapis.

Definicja w linii 44 pliku `generuj.cpp`.

4.1.4 Dokumentacja atrybutów składowych

4.1.4.1 `long int dane::rozmiar` `[private]`

Definicja w linii 26 pliku `generuj.hh`.

4.1.4.2 `long int* dane::tablica` `[private]`

Definicja w linii 25 pliku `generuj.hh`.

Dokumentacja dla tej klasy została wygenerowana z plików:

- [generuj.hh](#)
- [generuj.cpp](#)

5 Dokumentacja plików

5.1 Dokumentacja pliku `generuj.cpp`

```
#include <iostream>
#include <fstream>
#include <ctime>
#include "generuj.hh"
```

Funkcje

- `LARGE_INTEGER` [startTimer](#) ()
- `LARGE_INTEGER` [endTimer](#) ()

5.1.1 Dokumentacja funkcji

5.1.1.1 `LARGE_INTEGER endTimer ()`

Funkcja kończy pomiar czasu

Zwraca

koniec - zmierzony czas

Definicja w linii 14 pliku `generuj.cpp`.

5.1.1.2 LARGE_INTEGER startTimer ()

Funkcja rozpoczyna pomiar czasu.

Zwraca

start - zmierzony czas.

Definicja w linii 6 pliku generuj.cpp.

5.2 Dokumentacja pliku generuj.hh

```
#include <Windows.h>
```

Komponenty

- class `dane`

Funkcje

- LARGE_INTEGER `startTimer` ()
- LARGE_INTEGER `endTimer` ()

5.2.1 Dokumentacja funkcji

5.2.1.1 LARGE_INTEGER endTimer ()

Funkcja kończy pomiar czasu

Zwraca

koniec - zmierzony czas

Definicja w linii 14 pliku generuj.cpp.

5.2.1.2 LARGE_INTEGER startTimer ()

Funkcja rozpoczyna pomiar czasu.

Zwraca

start - zmierzony czas.

Definicja w linii 6 pliku generuj.cpp.

5.3 Dokumentacja pliku main.cpp

```
#include <iostream>
#include <windows.h>
#include <fstream>
#include "generuj.hh"
```

Funkcje

- `int main ()`

5.3.1 Dokumentacja funkcji

5.3.1.1 `int main ()`

Definicja w linii 26 pliku main.cpp.

Laboratorium 1 - Sprawozdanie

Wojciech Makuch

18 marca 2015

6 Zadanie

Napisać program generujący liczby pseudolosowe, a następnie liczący złożoność obliczeniową wymnożenia każdego z tych elementów przez 2.

7 Realizacja

Program zawiera jedną klasę przechowującą rozmiar oraz liczby pseudolosowe. Klasa zawiera metody wypełniające tablice, mnożące każdy z elementów przez 2, zliczając przy tym czas wykonania tej operacji. Ponadto program ma możliwość zapisania danych do pliku. Program nie posiada interfejsu z użytkownikiem.

8 Działanie

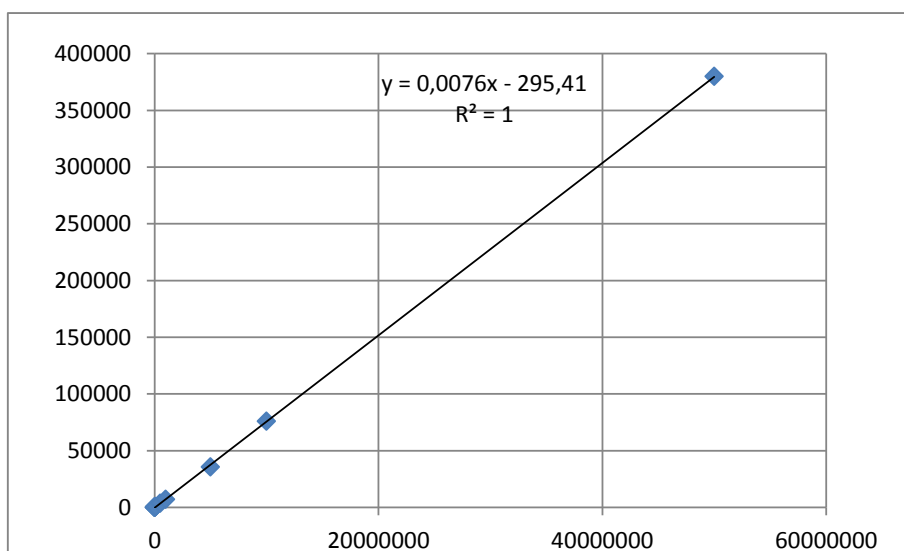
Główna funkcja programu w pętli wypełnia tablice liczbami pseudolosowymi z zakresu 10-10 000 000. Następnie zliczas czas wykonanych operacji i zapisuje w formie tabelki do pliku o nazwie Pomiar_czasu2.txt.

9 Wyniki

Rysunek1 pokazuje wykres zależności złożoności obliczeniowej w czasie. Z wykresu można zauważyć, że złożoność obliczeniowa jest liniowa, czyli $O(n)$.

10 Komentarz

Do utworzenia dokumentacji wykorzystano system Doxygen. Funkcja pomiaru czasu dla systemu Windows pobrana ze strony dr. J. Mierzwy. Program skompilowano w środowisku Code::Blocks. Do stworzenia wykresu posłużono się pakietem MS Excel, sprawozdanie napisano używając systemu \LaTeX .



Rysunek 1: Wykres złożoności obliczeniowej.