

Lab1 - mierzenie czasu wykonywania programu.

Generated by Doxygen 1.8.6

Sun Mar 15 2015 00:07:50

Contents

1	Main Page	1
1.1	Wykres przyrostu czasu od ilości danych wejściowych	2
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Class Documentation	7
4.1	DoubleNumbers Class Reference	7
4.1.1	Detailed Description	7
4.1.2	Constructor & Destructor Documentation	7
4.1.2.1	DoubleNumbers	7
4.1.3	Member Function Documentation	8
4.1.3.1	loadRandomNumbers	8
4.1.3.2	multiplyByTwo	8
4.2	Timer Class Reference	8
4.2.1	Detailed Description	8
4.2.2	Constructor & Destructor Documentation	8
4.2.2.1	Timer	8
4.2.3	Member Function Documentation	9
4.2.3.1	diffTimeMs	9
4.2.3.2	startTimer	9
4.2.3.3	stopTimer	9
5	File Documentation	11
5.1	inc/DoubleNumbers.h File Reference	11
5.1.1	Detailed Description	11
5.2	inc/timer.h File Reference	11
5.2.1	Detailed Description	11
5.3	src/DoubleNumbers.cpp File Reference	11
5.3.1	Detailed Description	12

5.4	src/main.cpp File Reference	12
5.4.1	Detailed Description	12
5.4.2	Function Documentation	12
5.4.2.1	main	12
5.5	src/timer.cpp File Reference	12
5.5.1	Detailed Description	13
Index		14

Chapter 1

Main Page

Czas wykonywania algorytmu wykonującego podstawową operację arytmetyczną.

Program realizuje mnożenie określonej liczby elementów tablicy przez dwa i wyznacza czas tej operacji.

Author

Mateusz Bencer

Date

2015.03.12

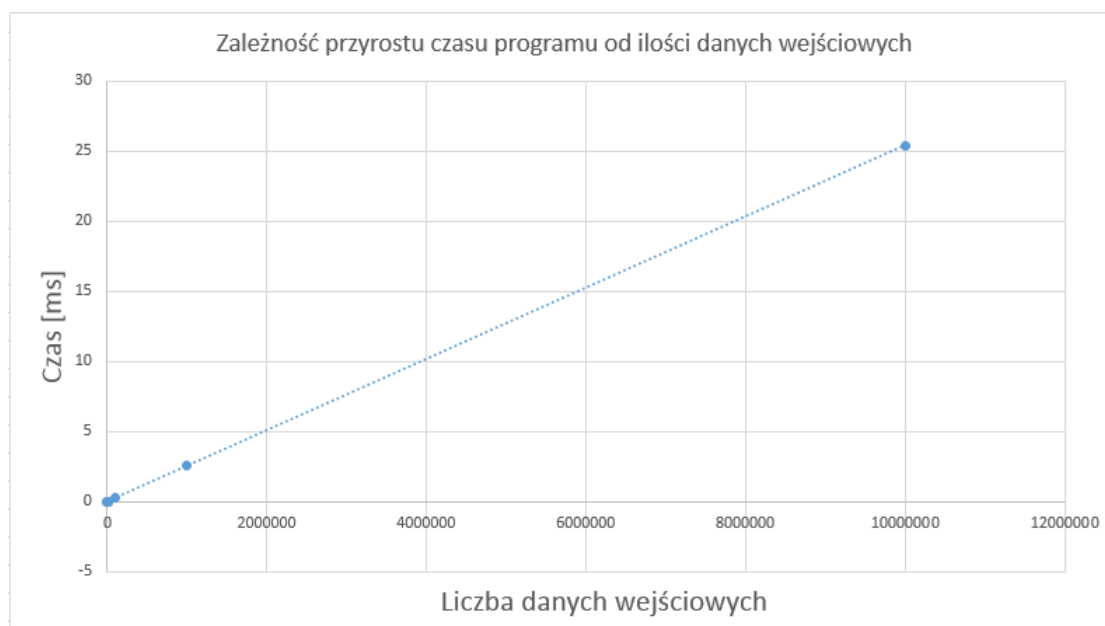
Version

1.0

Mail:

209360@pwr.wroc.edu.pl

1.1 Wykres przyrostu czasu od ilości danych wejściowych



Przebieg zależności czasu wykonywania programu od ilości danych wejściowych n zachowują w bardzo dużym przybliżeniu liniowy kształt. Na wykresie odzwierciedla to linia kreskowana, która jest linią trendu. Na tej podstawie możemy stwierdzić, że jest to algorytm o złożoności $O(n)$, co było proste do przewidzenia z powodu jednej, podstawowej operacji arytmetycznej powtarzanej w pętli n razy. Mnożenie przez dwa jest szczególnie szybką operacją dla procesora z powodu możliwości jej optymalizacji za pomocą przesunięcia bitowego w lewo. Największy istotny przyrost czasu można zauważyć przy liczbie danych wejściowych większych od 1 000 000. Sprawdzanie złożoności algorytmu metodą mierzenia czasu ma podstawową wadę: czas zależy od użytego sprzętu i obciążenia komputera w danym momencie. W związku z tym taki pomiar nie jest obiektywny w sensie ilościowym. Z drugiej jednak strony metoda ta jest znaczenie prostsza od liczenia podstawowych operacji wykonywanych przez procesor oraz pozwala przewidzieć zachowanie funkcji tempa wzrostu w zależności od ilości danych wejściowych.

Dla zwiększenia wiarygodności pomiary zostały wykonane 10 razy i uśrednione.

Wyniki otrzymane na moim komputerze:

$$10^0 : 0.0008 \quad (1.1)$$

$$10^1 : 0.0004 \quad (1.2)$$

$$10^2 : 0.0007 \quad (1.3)$$

$$10^3 : 0.003 \quad (1.4)$$

$$10^4 : 0.026 \quad (1.5)$$

$$10^5 : 0.255 \quad (1.6)$$

$$10^6 : 2.5345 \quad (1.7)$$

$$10^7 : 25.4004 \quad (1.8)$$

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

DoubleNumbers	Klasa do wykonania testowych obliczeń mnożenia elementów tablicy przez 2	7
Timer	Klasa do pomiaru różnicy czasów	8

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

inc/ DoubleNumbers.h	
Plik zawierający deklaracje klasy doubleNumbers	11
inc/ timer.h	
Plik zawierający deklaracje klasy timer służącej do pomiaru różnicy czasów	11
src/ DoubleNumbers.cpp	
Plik zawierający definicje funkcji klasy doubleNumbers	11
src/ main.cpp	
Plik zawierający sekwencje operacji do mierzenia czasu operacji mnożenia elementów tablicy przez 2	12
src/ timer.cpp	
Plik zawierający definicje funkcji klasy timer służącej do pomiaru różnicy czasów	12

Chapter 4

Class Documentation

4.1 DoubleNumbers Class Reference

Klasa do wykonania testowych obliczeń mnożenia elementów tablicy przez 2.

```
#include <DoubleNumbers.h>
```

Public Member Functions

- [DoubleNumbers](#) ()
Konstruktor domyślny ustawiający maksymalną liczbę tablicy na 10^6 oraz wczytujący czas systemowy potrzeby do losowania liczb.
- [DoubleNumbers](#) (unsigned long long size)
Konstruktor przyjmujący maksymalną wielkość tablicy oraz wczytujący czas systemowy potrzeby do losowania liczb.
- void [multiplyByTwo](#) (unsigned long long number)
Metoda wykonująca operację mnożenia przez 2 dla określonej liczby elementów tablicy.
- void [loadRandomNumbers](#) ()
Metoda wczytująca losowe liczby do tablicy.
- [~DoubleNumbers](#) ()
Destruktor zwalniający pamięć to tablicy `_numbers`.

4.1.1 Detailed Description

Klasa do wykonania testowych obliczeń mnożenia elementów tablicy przez 2.

Klasa pozwala na wylosowanie zadanej ilości liczb oraz pomnożenie przez 2 danej części z nich.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 DoubleNumbers::DoubleNumbers (unsigned long long size)

Konstruktor przyjmujący maksymalną wielkość tablicy oraz wczytujący czas systemowy potrzeby do losowania liczb.

Parameters

<i>size</i>	maksymalny rozmiar tablicy
-------------	----------------------------

4.1.3 Member Function Documentation

4.1.3.1 void DoubleNumbers::loadRandomNumbers ()

Metoda wczytująca losowe liczby do tablicy.

Precondition

Konieczna do poprawnego działania metody instrukcja srand jest zawarta w konstruktorze.

4.1.3.2 void DoubleNumbers::multiplyByTwo (unsigned long long *number*)

Metoda wykonująca operacje mnożenia przez 2 dla określonej liczby elementów tablicy.

Liczba mnożonych elementów tablicy jest determinowana przez parametr *number*.

Parameters

<i>number</i>	określa część tablicy poddanej operacji mnożenia przez 2.
---------------	---

The documentation for this class was generated from the following files:

- inc/[DoubleNumbers.h](#)
- src/[DoubleNumbers.cpp](#)

4.2 Timer Class Reference

Klasa do pomiaru różnicy czasów.

```
#include <timer.h>
```

Public Member Functions

- [Timer](#) ()
Konstruktor zerujący parametry.
- void [startTimer](#) ()
Zmierzenie czasu rozpoczęcia pomiaru.
- void [stopTimer](#) ()
Zmierzenie czasu zakończenia pomiaru.
- double [diffTimeMs](#) ()
Funkcja zwracająca różnicę czasu pomiędzy czasem rozpoczęcia i zakończenia pomiaru.

4.2.1 Detailed Description

Klasa do pomiaru różnicy czasów.

Klasa pozwala na pomiar czasów w danych momentach oraz na zwrócenie czasu, który upłynął pomiędzy tymi momentami

4.2.2 Constructor & Destructor Documentation

4.2.2.1 Timer::Timer ()

Konstruktor zerujący parametry.

Konstruktor ten odpowiada za zerowania zmiennych startu i stopu w celu możliwości późniejszego sprawdzenia, czy pomiary czasu konieczne do wyznaczenia różnicy zostały zrealizowane.

4.2.3 Member Function Documentation

4.2.3.1 `double Timer::diffTimeMs ()`

Funkcja zwracająca różnicę czasu pomiędzy czasem rozpoczęcia i zakończenia pomiaru.

Różnica czasu zwracana jest w milisekundach.

Precondition

Czas zakończenia pomiaru musi być większy (późniejszy) od czasu jego rozpoczęcia

Returns

Zwracana jest różnica czasu zrzutowana do typu `double`

4.2.3.2 `void Timer::startTimer ()`

Zmierzenie czasu rozpoczęcia pomiaru.

Funkcja zapamiętuje bieżący czas, jako czas rozpoczęcia pomiaru.

4.2.3.3 `void Timer::stopTimer ()`

Zmierzenie czasu zakończenia pomiaru.

Funkcja zapamiętuje bieżący czas, jako czas zakończenia pomiaru.

The documentation for this class was generated from the following files:

- [inc/timer.h](#)
- [src/timer.cpp](#)

Chapter 5

File Documentation

5.1 inc/DoubleNumbers.h File Reference

Plik zawierający deklaracje klasy doubleNumbers.

Classes

- class [DoubleNumbers](#)

Klasa do wykonania testowych obliczeń mnożenia elementów tablicy przez 2.

5.1.1 Detailed Description

Plik zawierający deklaracje klasy doubleNumbers. doubleNumbers.h

5.2 inc/timer.h File Reference

Plik zawierający deklaracje klasy timer służącej do pomiaru różnicy czasów.

```
#include <ctime>
```

Classes

- class [Timer](#)

Klasa do pomiaru różnicy czasów.

5.2.1 Detailed Description

Plik zawierający deklaracje klasy timer służącej do pomiaru różnicy czasów. [timer.h](#)

5.3 src/DoubleNumbers.cpp File Reference

Plik zawierający definicje funkcji klasy doubleNumbers.

```
#include "../inc/DoubleNumbers.h"
#include <stdlib.h>
#include <time.h>
```

5.3.1 Detailed Description

Plik zawierający definicje funkcji klasy doubleNumbers. doubleNumbers.cpp

5.4 src/main.cpp File Reference

Plik zawierający sekwencje operacji do mierzenia czasu operacji mnożenia elementów tablicy przez 2.

```
#include <iostream>
#include "../inc/timer.h"
#include "../inc/DoubleNumbers.h"
#include <math.h>
```

Functions

- int [main](#) ()

5.4.1 Detailed Description

Plik zawierający sekwencje operacji do mierzenia czasu operacji mnożenia elementów tablicy przez 2. [main.cpp](#)

5.4.2 Function Documentation

5.4.2.1 int main ()

liczba określająca z jakiej liczby pomiarów będzie brana średnia

Wykładnik rozmiaru problemu

Maksymalny rozmiar tablicy do testów

Wielkości rozmiarów tablic, dla których trzeba zmierzyć czas obliczeń

Czasy wykonania obliczeń dla danych wielkości tablicy.

Obiekt do pomiaru różnicy czasu.

Obiekt do wykonywania operacji mnożenia tablicy przez 2.

5.5 src/timer.cpp File Reference

Plik zawierający definicje funkcji klasy timer służącej do pomiaru różnicy czasów.

```
#include "../inc/timer.h"
#include <iostream>
```


5.5.1 Detailed Description

Plik zawierający definicje funkcji klasy timer służącej do pomiaru różnicy czasów. [timer.cpp](#)

Index

- diffTimeMs
 - Timer, [9](#)
- DoubleNumbers, [7](#)
 - DoubleNumbers, [7](#)
 - DoubleNumbers, [7](#)
 - loadRandomNumbers, [8](#)
 - multiplyByTwo, [8](#)
- inc/DoubleNumbers.h, [11](#)
- inc/timer.h, [11](#)
- loadRandomNumbers
 - DoubleNumbers, [8](#)
- main
 - main.cpp, [12](#)
- main.cpp
 - main, [12](#)
- multiplyByTwo
 - DoubleNumbers, [8](#)
- src/DoubleNumbers.cpp, [11](#)
- src/main.cpp, [12](#)
- src/timer.cpp, [12](#)
- startTimer
 - Timer, [9](#)
- stopTimer
 - Timer, [9](#)
- Timer, [8](#)
 - diffTimeMs, [9](#)
 - startTimer, [9](#)
 - stopTimer, [9](#)
 - Timer, [8](#)