

PAMSI 1

Wygenerowano przez Doxygen 1.8.6

Pn, 10 mar 2014 01:10:19

## Spis treści

<b>1</b>	<b>Strona główna</b>	<b>1</b>
<b>2</b>	<b>Indeks plików</b>	<b>1</b>
2.1	Lista plików . . . . .	1
<b>3</b>	<b>Dokumentacja plików</b>	<b>1</b>
3.1	Dokumentacja pliku TestAlgorytmu.cpp . . . . .	1
3.1.1	Dokumentacja funkcji . . . . .	2
<b>4</b>	<b>Sprawozdanie z wykonania programu</b>	<b>3</b>
	<b>Indeks</b>	<b>5</b>

## 1 Strona główna

Laboratorium 1.

Funkcje do pomiaru czasu wykonywania algorytmu. Algorytm mnoży każdą liczbę przez 2.

Autor

Jakub Chmiel 200314

## 2 Indeks plików

### 2.1 Lista plików

Tutaj znajduje się lista wszystkich udokumentowanych plików z ich krótkimi opisami:

<b>TestAlgorytmu.cpp</b>	<b>1</b>
--------------------------	----------

## 3 Dokumentacja plików

### 3.1 Dokumentacja pliku TestAlgorytmu.cpp

```
#include <iostream>
#include <fstream>
#include <time.h>
#include <stdlib.h>
```

Definicje

- #define **PLIK\_DANYCH** "dane1.txt"  
*nazwa pliku z danymi wejściowymi, także dla wyjścia generowanych liczb.*
- #define **PLIK\_WYJSCIOWY** "wyjscie1.csv"  
*nazwa pliku wyjściowego.*
- #define **LICZBA\_POWTORZEN** 50

- ilosc powtorzen pomiaru czasu dla kazdego rozmiaru problemu.*
- #define `LICZBA_WIELKOSCI` 8  
*ilosc roznych rozmiarow problemu.*
- #define `WIELKOSCI_PROBLEMU` {10, 1000, 100000, 1000000, 2000000, 4000000, 6000000, 10000000}  
*tablica zawierajaca wszystkie mierzone rozmiary problemu.*
- #define `WIELKOSC_GENEROWANYCH_DANYCH` 10000000  
*ilosc danych do wygenerowania.*
- #define `TYP` int  
*typ danych.*

## Funkcje

- void `algorytm` (`TYP` \*tablica, int rozmiar)  
*Testowany algorytm.*
- bool `wczytaj_dane` (const char \*nazwa\_pliku, `TYP` \*&tablica, int &rozmiar)  
*wczytuje dane z pliku*
- bool `zapisz_dane` (const char \*nazwa\_pliku, int \*col\_rozmiar\_problemu, double \*col\_czas, int rozmiar)  
*zapisuje dane do pliku .csv dane zawieraja: 1 kolumna: wielkosc problemu 2 kolumna: czas potrzebny do zrealizowania danego problemu*
- void `wypisz_dane` (`TYP` \*tablica, int rozmiar)  
*wypisuje tablice na standardowe wyjscie*
- void `kopiuj` (`TYP` \*tab\_z, `TYP` \*tab\_do, int ile)  
*kopiuje n elementow tablicy do innej*
- void `testuj_algorytm` (`TYP` \*tablica, int rozmiar)  
*wykonuje testy czasu algorytmu dla przygotowanych parametrow zmierzone czasu zapisuje do pliku*
- void `generuj_dane` ()  
*generuje dane jesli ich nie ma*
- int `main` ()  
*funkcja main*

### 3.1.1 Dokumentacja funkcji

#### 3.1.1.1 void `algorytm` ( `TYP` \* *tablica*, int *rozmiar* )

Testowany algorytm.

##### Parametry

<i>*tablica</i>	dane do przetworzenia
<i>rozmiar</i>	ilosc danych

#### 3.1.1.2 void `kopiuj` ( `TYP` \* *tab\_z*, `TYP` \* *tab\_do*, int *ile* )

kopiuje n elementow tablicy do innej

##### Parametry

<i>*tab_z</i>	tablica wyjsciowa
<i>*tab_do</i>	tablica docelowa
<i>ile</i>	ilosc elementow do skopiowania

#### 3.1.1.3 void `testuj_algorytm` ( `TYP` \* *tablica*, int *rozmiar* )

wykonuje testy czasu algorytmu dla przygotowanych parametrow zmierzone czasu zapisuje do pliku

## Parametry

<i>*tablica</i>	dane
<i>rozmiar</i>	rozmiar tablicy

## 3.1.1.4 bool wczytaj\_dane ( const char \* nazwa\_pliku, TYP \*&amp; tablica, int &amp; rozmiar )

wczytuje dane z pliku

Format:

liczba\_danych

dana1

dana2

.

## Parametry

<i>*nazwa_pliku</i>	nazwa pliku z danymi
<i>*&amp;tablica</i>	tablica docelowa (usuwana w przypadku !=NULL)
<i>&amp;rozmiar</i>	rozmiar tablicy docelowej

## 3.1.1.5 void wypisz\_dane ( TYP \* tablica, int rozmiar )

wypisuje tablice na standardowe wyjście

## Parametry

<i>*tablica</i>	tablica
<i>rozmiar</i>	rozmiar tablicy

## 3.1.1.6 bool zapisz\_dane ( const char \* nazwa\_pliku, int \* col\_rozmiar\_problemu, double \* col\_czas, int rozmiar )

zapisuje dane do pliku .csv dane zawieraja: 1 kolumna: wielkosc problemu 2 kolumna: czas potrzebny do zrealizowanego danego problemu

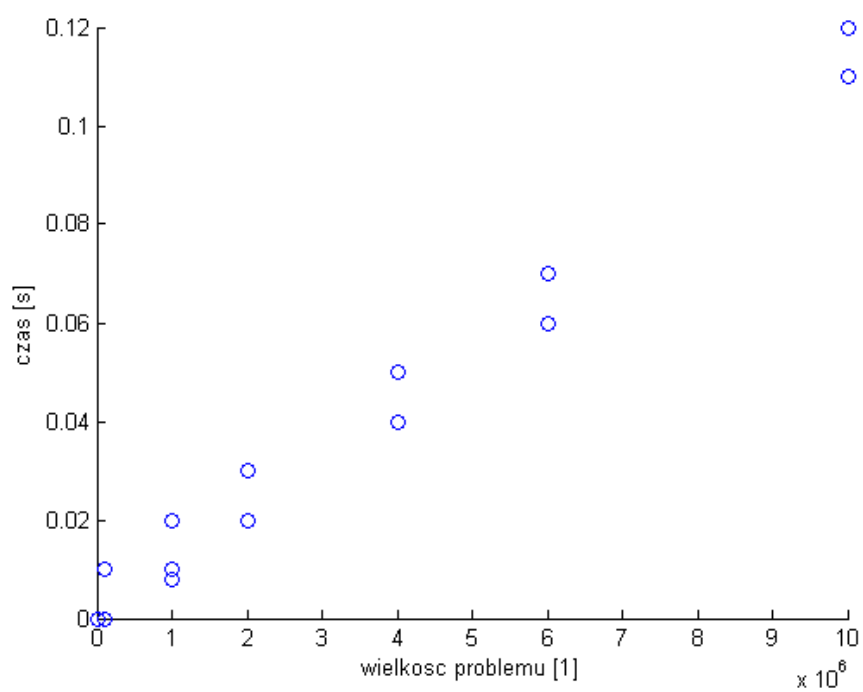
## Parametry

<i>*nazwa_pliku</i>	nazwa pliku do zapisu
<i>*col_rozmiar_ - problemu</i>	tablica z 1 kolumna
<i>*col_czas</i>	druga kolumna
<i>rozmiar</i>	rozmiar obu tablic

## Zwraca

- true sukces
- false blad

## 4 Sprawozdanie z wykonania programu



Rysunek 1: Wykres pomiaru czasu dla problemu różnego rozmiaru

## Skorowidz

algorytm

TestAlgorytmu.cpp, [2](#)

kopiuj

TestAlgorytmu.cpp, [2](#)

TestAlgorytmu.cpp, [1](#)

algorytm, [2](#)

kopiuj, [2](#)

testuj\_algorytm, [2](#)

wczytaj\_dane, [3](#)

wypisz\_dane, [3](#)

zapisz\_dane, [3](#)

testuj\_algorytm

TestAlgorytmu.cpp, [2](#)

wczytaj\_dane

TestAlgorytmu.cpp, [3](#)

wypisz\_dane

TestAlgorytmu.cpp, [3](#)

zapisz\_dane

TestAlgorytmu.cpp, [3](#)