

My Project

Wygenerowano przez Doxygen 1.7.6.1

Fri Mar 21 2014 12:46:05

Spis treści

Rozdział 1

Stos za pomoca tablicy ver.2

Autor

Krzysztof Kucharczyk

Data

15.03.2014

Wersja

3

Program modelujący zachowanie stosu na bazie tablicy dwówymiarowej.

Charakterystycznym elementem jest zwiększanie tablicy dwukrotnie w momencie, gdy miałaby być przepełniona i zwalnianiu połowy miejsca, gdy ilość elementów wynosi $1/4$ pojemności tablicy.

Rozdział 2

Indeks klas

2.1 Lista klas

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

[Stos](#)

Modeluje pojęcie stosu ??

Rozdział 3

Indeks plików

3.1 Lista plików

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

/home/krzysztof/Desktop/PAMSI/Laboratorium/Laboratorium_3/Stos_-
t2/inc/[stos.h](#)

Plik nagłówkowy elementu stos ??

Rozdział 4

Dokumentacja klas

4.1 Dokumentacja klasy Stos

Modeluje pojęcie stosu.

```
#include <stos.h>
```

Metody publiczne

- **Stos** (int wielkosc)
- void **push** (int Nowy)
Dodaje element na stos.
- void **pop** ()
Zdejmuje element ze stosu.
- bool **isEmpty** ()
Sprawdza, czy stos jest pusty.
- int **size** ()
Zwraca ilość elementów na stosie.
- void **Show** ()
Wyświetla zawartość stosu.
- int * **Pobierz_dane** ()
Pobiera dane z pliku.
- void **Eksportuj_dane** (double czas)
Umożliwia eksportowanie danych do pliku.
- int **Opcje** (int argc, char **argv)

Atrybuty publiczne

- int * **Tablica**
Wskaźnik na tablicę danych.

- int `wielkosc_tablicy`
Określa aktualną wielkość tablicy.
- int `ilosc_elementow`
- int `zakres`
Określa zakres pobierania danych.

4.1.1 Opis szczegółowy

Modeluje pojęcie stosu.

Klasa moduluje pojęcie stosu jako konkretnej realizacji tablicy dynamicznej. Sytuacja przeciążenia stosu rozwiązana jest poprzez tworzenie nowych, o jeden element większych tablic dynamicznych. Analogicznie, zwalnianie pamięci następuje każdorazowo po usunięciu elementu.

4.1.2 Dokumentacja funkcji składowych

4.1.2.1 void `Stos::Eksportuj_dane` (double `czas`)

Umożliwia eksportowanie danych do pliku.

Metoda umożliwia eksportowanie danych związanych z obliczaniem czasu działania algorytmu do pliku o nazwie "Wyniki_temp.txt".

Parametry

<code>in</code>	<code>czas</code>	- długość wykonywania operacji w nanosekundach
-----------------	-------------------	--

4.1.2.2 bool `Stos::isEmpty` ()

Sprawdza, czy stos jest pusty.

Metoda określa, czy stos jest pusty. Zwraca 1 gdy stos jest pusty i 0 gdy nie pusty.

4.1.2.3 int * `Stos::Pobierz_dane` ()

Pobiera dane z pliku.

Metoda pobiera z pliku "Zakres_temp.txt" zmienną określającą ilość danych, następnie tworzy dynamiczną tablicę o wielkości równej zakresowi i ostatecznie otwiera plik "-Dane_testowe.txt", z którego pobiera "zakres" danych. Wskaźnik na tablicę z danymi zostaje zwrócony.

Zwraca

Wskaźnik na tablicę z danymi

4.1.2.4 void Stos::pop ()

Zdejmuje element ze stosu.

Metoda zdejmuję element znajdujący się aktualnie na stosie. Tablica zostaje nadpisana nową, o 1 mniejszą tablicą, a stara zostaje zwolniona. Brak elementów do usunięcia zostaje wyszczególniony stosownym komunikatem.

4.1.2.5 void Stos::push (int Nowy)

Dodaje element na stos.

Metoda służy do dodania elementu na szczyt stosu. W przypadku przepełnienia stosu tworzy nową, większą o 1 tablicę i przepisuje o niej dane, by ostatecznie dodać nowy element. Stara tablica zostaje zwolniona, a wskaźnik nadpisany.

Parametry

in	Nowy	- element, który ma zostać dodany na szczyt stosu
----	------	---

4.1.2.6 void Stos::Show ()

Wyświetla zawartość stosu.

Metoda pomocnicza, umożliwiająca śledzenia zmian zachodzących na stosie poprzez wyświetlenie jego zawartości.

4.1.2.7 int Stos::size ()

Zwraca ilość elementów na stosie.

Metoda zwraca aktualną ilość elementów na stosie.

Zwraca

Ilość elementów

4.1.3 Dokumentacja atrybutów składowych

4.1.3.1 int Stos::ilosc_elementow

Określa ilość elementów na stosie.

4.1.3.2 int* Stos::Tablica

Wskaźnik na tablicę danych.

Wskazuje na strukturę zawierającą dane i na której odbywają się wszelkie operacje.

4.1.3.3 int Stos::wielkosc_tablicy

Określa aktualną wielkość tablicy.

Określa wielkość tablicy, co przydaje się przy operacji zwiększania/zmniejszania tablicy

4.1.3.4 int Stos::zakres

Określa zakres pobierania danych.

Wskazuje ile danych ma zostać pobranych z pliku z danymi. Innymi słowy, określa wielkość problemu, z jakim algorytm ma się uporać.

Dokumentacja dla tej klasy została wygenerowana z plików:

- /home/krzysztof/Desktop/PAMSI/Laboratorium/Laboratorium_3/Stos_t2/inc/[stos.h](#)
- /home/krzysztof/Desktop/PAMSI/Laboratorium/Laboratorium_3/Stos_t2/src/stos.cpp

Rozdział 5

Dokumentacja plików

5.1 Dokumentacja pliku /home/krzysztof/Desktop/PAMSI/Laboratorium/-Laboratorium_3/Stos_t2/inc/stos.h

Plik nagłówkowy elementu stos.

```
#include <iostream> #include <fstream> #include <cstdlib> x
```

Komponenty

- class [Stos](#)
Modeluje pojęcie stosu.

5.1.1 Opis szczegółowy

Plik nagłówkowy elementu stos. Plik zawiera definicje klasy [Stos](#).