



STOS

LINIOWA STRUKTURA DANYCH

SPIIS TREŚCI

- Budowa
- Działania na stosie
- Implementacja C++
- `std::stack`
- Przykłady zastosowań

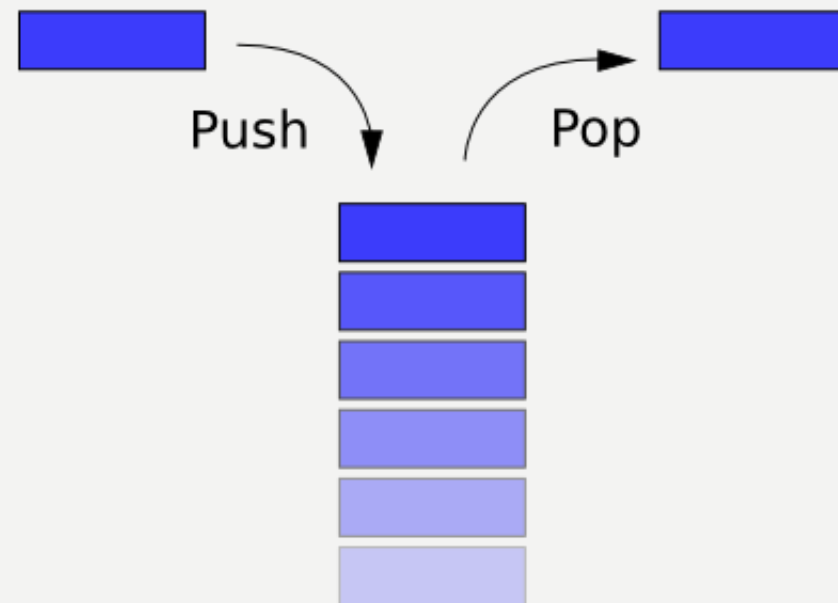


BUDOWA

- Wskaźnik na tablicę elementów
- Pole top (miejsce elementu na szczycie)
- Pojemność stosu

DZIAŁANIA NA STOSIE

- Dodaj (Push) – dodaje element na szczyt stosu
- Usuń (Pop) – usuwa element ze szczytu stosu
- czyPusty (isEmpty) – sprawdza czy stos nie jest pusty
- Wyświetl (Print) – wyświetla stos w kolejności LIFO



IMPLEMENTACJA C++

- Tablica statyczna
- Wektor
- Lista
- Za pomocą wskaźników

PRZYKŁAD – TABLICA STATYCZNA

```
template<typename T>
class Stack {
    //Pola
private:
    T *p;                //Gdzie p będzie wskazywać na tablicę statyczną
    int top;
    int capacity;        //Pojemność stosu

    //Konstruktory i dsetruktor
public:
    Stack(const int& size) {...}
    ~Stack(){ if (p != 0) delete[]p;}

    //Metody
    void push(const T& elem) {...}
    void pop() {...}
    bool isEmpty() {...}
    void erase() {...}
    void print() {...}
    const T& topEl() {...}
};
```

STD::STACK

std::stack – gotowa implementacja stosu zawarta w bibliotece standardowej

Operacje:

- bool empty() - zwraca wartość typu bool, jeśli pusty true, w innym wypadku false
- void pop() - usuwa szczytowy element z niepustego stosu
- void push(const value_type& value) - wstawia element na szczyt stosu
- value_type& top() – zwraca referencję do szczytowego (ostatniego) element
- size_type size() – zwraca liczbę elementów na stosie
- void swap(stack& other) – zamienia zawartość dwóch stosów ze sobą

STD::STACK – PRZYKŁADOWY KOD 1

```
#include <Windows.h>
#include <iostream>
#include <stack>
int main(){
system("chcp 1250");//Umożliwia polskie litery w konsoli
system("cls");//Czyści konsolę

std::stack<int> stos;

int n, randTemp;
std::cout << "Podaj n: "; std::cin >> n;

std::cout << "\nNapełnianie stosu:\n";
for (int i = 1; i <= n; i++) {
randTemp = rand() % 100;
std::cout << randTemp << "\n";
stos.push(randTemp); //Wypełniamy stos losowymi liczbami
}

std::cout << "\nZawartość stosu:\n";
for (int i = 1; i <= n; i++) {
std::cout << stos.top() << "\n";
stos.pop();//Aby wyświetlić wszystkie elementy musimy "rozebrać" stos
}

std::cout << "\n" << (stos.empty() ? "Stos pusty" : "Stos nie pusty") << "\n";

system("pause");//Zbędne jeśli włączamy program bezpośrednio z exe
exit (0);}
```

Wynik w konsoli:

Podaj n: 10

Napełnianie stosu:

41
67
34
0
69
24
78
58
62
64

Zawartość stosu:

64
62
58
78
24
69
0
34
67
41

Stos pusty
Press any key to continue . . .

STD::STACK – PRZYKŁADOWY KOD 2

```
int main(){
    system("chcp 1250");
    system("cls");

    std::stack<int> stos1, stos2;

    int n, randTemp;
    std::cout << "Podaj n dla stosu 1: "; std::cin >> n;

    std::cout << "Napełnianie stosu 1\n";
    for (int i = 1; i <= n; i++) stos1.push(rand() % 100);

    std::cout << "\nPodaj n dla stosu 2: "; std::cin >> n;

    std::cout << "Napełnianie stosu 2\n";
    for (int i = 1; i <= n; i++) stos2.push(rand() % 100);

    std::cout << "\nRozmiar stosu 1: " << stos1.size() << "\n";
    std::cout << "Szczytowy element stosu 1: " << stos1.top() << "\n";
    std::cout << "Rozmiar stosu 2: " << stos2.size() << "\n";
    std::cout << "Szczytowy element stosu 2: " << stos2.top() << "\n";

    stos1.swap(stos2); //Zamienia ze sobą zawartość stosów
    std::cout << "\nRozmiar stosu 1: " << stos1.size() << "\n";
    std::cout << "Szczytowy element stosu 1: " << stos1.top() << "\n";
    std::cout << "Rozmiar stosu 2: " << stos2.size() << "\n";
    std::cout << "Szczytowy element stosu 2: " << stos2.top() << "\n\n";

    system("pause");//Zbędne jeśli włączamy program bezpośrednio z exe
    exit (0);
}
```

Wynik w konsoli:

Podaj n dla stosu 1: 10
Napełnianie stosu 1

Podaj n dla stosu 2: 15
Napełnianie stosu 2

Rozmiar stosu 1: 10
Szczytowy element stosu 1: 64
Rozmiar stosu 2: 15
Szczytowy element stosu 2: 92

Rozmiar stosu 1: 15
Szczytowy element stosu 1: 92
Rozmiar stosu 2: 10
Szczytowy element stosu 2: 64

Press any key to continue . . .

PRZYKŁADY ZASTOSOWAŃ

- [Konwersja liczby binarnej na dziesiętną](#)
- [Obliczanie wyrażeń w ONP](#)
- [Procesor \(adresy powrotów\)](#)

Notacja tradycyjna	ONP
$2+3$	2 3 +
$4 \cdot (2 + 3)$	4 2 3 + ·
$(2 + 3) \cdot 4$	2 3 + 4 ·
$2 \cdot 3 - 4 \cdot 5$	2 3 · 4 5 · -

ŹRÓDŁA

- http://achilles.tu.kielce.pl/Members/achrobot/archiwum/semestr-ii-2015-2016/pdf/pp/pp2_lecture_2.pdf
- [https://pl.wikipedia.org/wiki/Stos_\(informatyka\)](https://pl.wikipedia.org/wiki/Stos_(informatyka))
- http://www.informatyka.orawskie.pl/?pl_stos,101 (Grafiki)
- <http://www.cplusplus.com/reference/stack/stack/>
- <http://en.cppreference.com/w/cpp/container/stack>



KONIEC