

Programowanie

Dariusz Wardecki, VIII

Powtórzenie

Co wypisze program?

```
#include<iostream>

using namespace std;

int main (int argc, char *argv[])
{
    double a = argv[1][0];
    double b = argv[2][0];
    double c = a + b;
    cout << c << endl;
    return 0;
}
```

```
./program 5 6
```

Powtórzenie

Popraw błąd

```
#include<iostream>

using namespace std;

int main (int argc, char *argv[])
{
    double tab[] = {32, 345.65, 33, 1234};
    double *w = tab + *argv[1];
    cout << w << " " << *w << endl;
    return 0;
}
```

Struktury

```
struct moja_struktura
{
    typ zmienna1;
    typ zmienna2;
    ...
};
main( )
{
    moja_struktura obiekt;
    ...
}
```

Struktury

```
struct osoba
{
    string imie;
    int wiek;
};
main( )
{
    osoba student;
    student.imie = "Paweł";
    student.wiek = 20;
}
```

Struktury

```
struct osoba
{ ... };
main( )
{
    osoba student;
    osoba *wsk = &student;
    wsk->imie = "Paweł";
    wsk->wiek = 20;
}
```

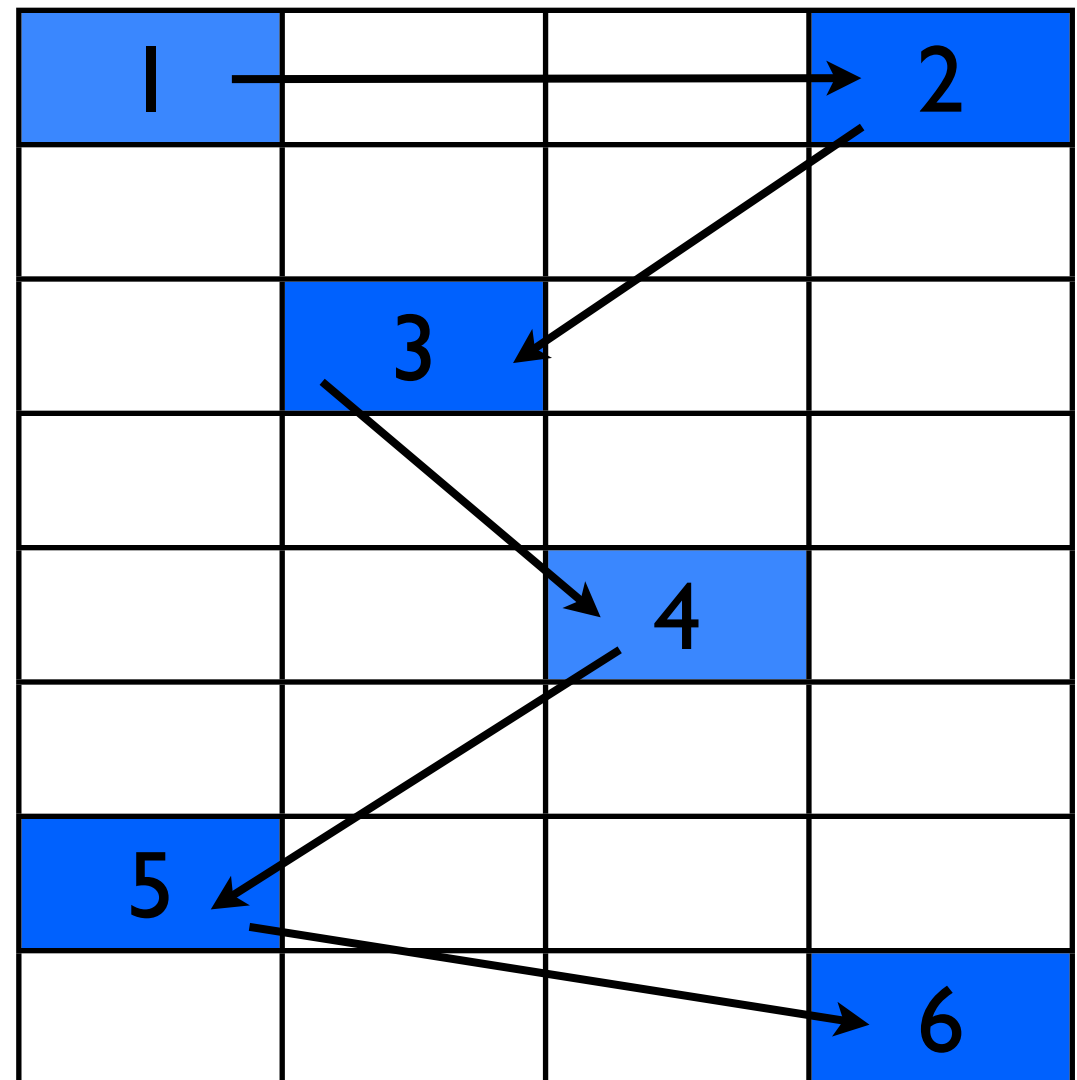
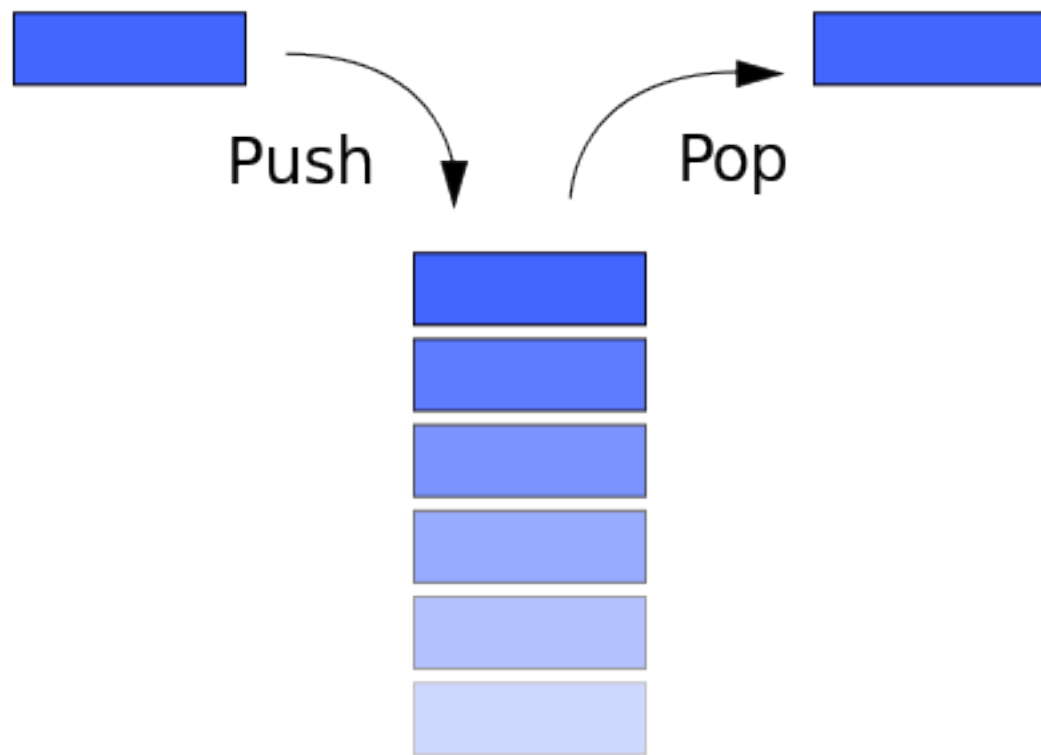
Struktury

```
struct osoba
{ ... };
main( )
{
    osoba *wsk = new osoba;

    wsk->imie = "Paweł";
    wsk->wiek = 20;
}
```

Stos

przestrzeń adresów



Stos

Można użyć struktury:

```
struct Element{  
    double wartosc;  
    Element *poprzedni;  
};
```

Element 2 (add 2)
2.5
add 1

Element 1 (add 1)
5
0

Stos

Jak dokładać i zdejmować ze stosu?

```
Element *top = 0;
```

```
void push (double wart)
{
    Element *tmp = top;
    top = new Element;
    top->wartosc = wart;
    top->poprzedni = tmp;
}
```

Stos

Jak dokładać i zdejmować ze stosu?

```
int pop (double &wart)
{
    if (!top)
        return 0;
    wart=top->wartosc;
    Element *tmp = top->poprzedni;
    delete top;
    top=tmp;
    return 1;
}
```

Stos

Jak dokładać i zdejmować ze stosu?

```
int main ()
{
    double liczba;
    while (cin>>liczba) push (liczba);
    while (pop (liczba))
        cout << liczba <<" ";
    cout<<endl;
    return 0;
}
```