

Programowanie

Dariusz Wardecki, wyk.VI

Powtórzenie

Co wypisze program?

```
#include<iostream >
using namespace std;

int main ()
{
    int a = 2, b = 7;
    cout << a << " " << b << endl;
    int *p = &a;
    cout << p << " " << *p << endl;
    *p += b;
    cout << p << " " << *p << endl;
    p = &b;
    cout << p << " " << *p << endl;
    cout << a << " " << b << endl;
    return 0;
}
```

Powtórzenie

Co wypisze program?

```
#include<iostream >
using namespace std;

int main ()
{
    int tab[4] = {34, 65, 44, 21};
    cout << tab << " " << tab[0] << endl;
    int *p = tab;
    cout << p << " " << *p << endl;
    *(p++) += 10;
    *(++p) += 10;
    for(int i = 0; i < 4; i++){
        cout << tab[i] << endl;
    }
    return 0;
}
```

Sortowanie tablic

Sortowanie bąbelkowe

Tablica wejściowa: (5 , 2 , 43 , 1 , 22)

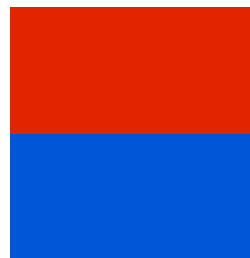
iteracja 1

iteracja 2

iteracja 3

iteracja 4

(5 , 2 , 43 , 1 , 22)	(2 , 5 , 1 , 22 , 43)	(2 , 1 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)
(2 , 5 , 43 , 1 , 22)	(2 , 5 , 1 , 22 , 43)	(1 , 2 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)
(2 , 5 , 43 , 1 , 22)	(2 , 1 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)
(2 , 5 , 1 , 43 , 22)	(2 , 1 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)	(1 , 2 , 5 , 22 , 43)



zamiana miejscami

brak zamiany

Sortowanie tablic

Sortowanie przez selekcję

Tablica wejściowa: (5 , 22 , 43 , 1 , 2)

iteracja 1

iteracja 2

iteracja 3

iteracja 4

(5 , 22 , 43 , 1 , 2)	(1 , 22 , 43 , 5 , 2)	(1 , 2 , 43 , 5 , 22)	(1 , 2 , 5 , 43 , 22)
(5 , 22 , 43 , 1 , 2)	(1 , 22 , 43 , 5 , 2)	(1 , 2 , 43 , 5 , 22)	(1 , 2 , 5 , 43 , 22)
(1 , 22 , 43 , 5 , 2)	(1 , 2 , 43 , 5 , 22)	(1 , 2 , 5 , 43 , 22)	(1 , 2 , 5 , 22 , 43)



zamiana miejscami
przeszukiwany obszar
najmniejszy element

Łańcuchy (C-string)

Przykład C-string

```
char tab[14] = "Ala ma kota.";
```

A	l	a		m	a		k	o	t	a	.	\0	
---	---	---	--	---	---	--	---	---	---	---	---	----	--

65	108	97	32	109	97	32	107	111	116	97	46	0	
----	-----	----	----	-----	----	----	-----	-----	-----	----	----	---	--

Łańcuchy (C-string)

Przykład C-string

```
char tab[14] = "Ala ma kota.";
```

```
char tab[14] = {'A', 'l', 'a', ' ', 'm', 'a',  
' ', 'k', 'o', 't', 'a', '.', '\0'};
```

```
char tab[14] = {65, 108, 97, 32, 109,  
97, 32, 107, 111, 116, 97, 46, 0};
```

Łańcuchy (C-string)

```
#include <iostream>
using namespace std;

int main ()
{
    char pytanie[] = "Podaj imie: ";
    char pozdr[] = "Witaj, ";
    char imie[80];
    cout << pytanie;
    cin >> imie;
    cout << pozdr << imie << "!";
    return 0;
}
```

Łańcuchy (C-string)

C-string. Zamiana małych liter na wielkie

```
void zamien(char* t)
{
    int i = 0;
    while(t[i])
    {
        if(t[i]>=97 && t[i]<=122)
            t[i] = t[i] - 32;
        i++;
    }
}
```

Łańcuchy (C-string)

C-string. Długość łańcucha

```
int length(char* t)
{
    int i = 0;
    while(t[i]) i++;
    return i;
}
```

Argumenty funkcji main

Przykład

```
#include<iostream>
using namespace std;

int main(int argc, char* argv[ ])
{
    cout << "Liczba argumentow: " << argc << endl;
    cout << "Nazwa programu: " << argv[0] << endl;
    cout << "Argument pierwszy: " << argv[1] << endl;
    cout << "Argument drugi: " << argv[2] << endl;
    // itd ...
    return 0;
}
```

Klasa string

Przykład

```
#include<iostream>
#include<string>
using namespace std;

int main ()
{
    string str = "Hello";
    cout << str << endl;
    return 0;
}
```

Klasa string

Przykład

```
#include<iostream>
#include<string>
using namespace std;

int main ()
{
    string str1 = "Hello ";
    string str2 = "World";
    cout << str1 + str2 << endl;
    return 0;
}
```

Klasa string

Operacje na łańcuchach - insert

```
#include<iostream>
#include<string>
using namespace std;

int main ()
{
    string str1 = "Raz dwa cztery";
    string str2 = " trzy";
    str1.insert(7,str2);
    cout << str1 << endl;
    return 0;
}
```

Klasa string

Operacje na łańcuchach

- erase

```
#include<iostream>
#include<string>
using namespace std;

int main ()
{
    string str1 = "Raz dwa cztery";
    string str2 = " trzy";
    str1.erase(8,6);
    cout << str1 << endl;
    return 0;
}
```