



Wykład: 11

Trochę różnych przykładów



Przykłady

Przykład I - palindrom

Program sprawdza, czy podany wyraz jest palindromem.

```
1  #include <cstdlib>
2  #include <iostream>
3  #include <string>
4  using namespace std;
5
6  bool palindrom(string wyraz);
7
8  int main()
9  {
10     string wyraz;
11     cout << "Podaj wyraz do sprawdzenia:" << endl;
12     cin >> wyraz;
13     if (palindrom(wyraz)) cout << "TAK" << endl;
14     else cout << "NIE"
15     return 0;
16 }
17
18 bool palindrom(string wyraz)
19 {
20     string odwrocony;
21     int dlugosc = wyraz.length();
22     for(int i=0; i<dlugosc; i++)
23         odwrocony+=wyraz[dlugosc-i-1];
24     return wyraz==odwrocony;
25 }
```

Palindromem nazywamy wyraz, który czytany z lewej do prawej jest taki sam jak z prawej do lewej np.:

kajak
kobyłamamałybok

Przykład I – palindrom (inny sposób)

```
1 // program sprawdzający czy dany wyraz jest palindromem
2 // www.algorytm.org
3 #include<iostream>
4 using namespace std;
5 #include<string>
6
7 int main()
8 {
9     int i,j;
10    string wyraz;
11    cout << "Podaj wyraz, który chcesz sprawdzić: ";
12    cin >> wyraz;
13    // i - pokazuje na kolejne litery wyrazu poczynawszy od początku, j - zaczyna od końca
14    // funkcja length() wywołana na obiekcie string zwraca jego długość
15    for (i = 0, j = wyraz.length()-1; i < j; i++, j--)
16    {
17        if (wyraz[i] != wyraz[j])
18            break;
19    }
20    if (i < j)
21        cout << "Podany wyraz nie jest palindromem" << endl;
22    else
23        cout << "Podany wyraz jest palindromem" << endl;
24
25    return 0;
26 }
```

Źródło: www.algorytm.org

Przykład 2 – Numerowanie linii w pliku

```
1  #include <iostream>
2  #include <fstream>
3
4  using namespace std;
5
6  int main()
7  {
8      string tablica[100], s;
9      int n=0;
10     fstream plik;
11     plik.open("kod.cpp", ios::in);
12     while (!plik.eof())
13     {
14         getline(plik,s);
15         if (s.length()>0)
16             tablica[n++]=s;
17     }
18     plik.close();
19     plik.open("kod.cpp", ios::out);
20     for (int i=0; i<n; i++) {
21         cout << tablica[i] << endl;
22         plik << i << ": " << tablica[i] << endl;
23     }
24     plik.close();
25     return 0;
26 }
```

Program numeruje linie w pliku kod.cpp (przy okazji usuwając linie pste)

Przykład 3 – wyodrębnienie wyrazów z pliku

```
1  #include <iostream>
2  #include <fstream>
3  using namespace std;
4  int main()
5  {
6      string wyrazy[300], temp,s;
7      fstream plik;
8      int p, lw=0;
9      plik.open("dane.txt", ios::in);
10     if (!plik.good()){
11         cout <<"brak pliku";
12         return 0;
13     }
14     while (!plik.eof())
15     {
16         getline(plik, temp);
17         cout<< temp<<endl;
18         p=0;
19         while (p<temp.length())
20         { //w pobranym lancuchu przeskakuje znaki
39     }
40     //wypisuje tablice wyrazow
41     cout<<endl<<"-----"<<endl;
42     for (int i =0; i<lw;i++)
43         cout <<wyrazy[i] <<"-";
44     return 0;
45 }
```

- Program pobiera kolejno wszystkie linie z pliku, aż natrafi na jego koniec.
- Następnie pętla while() – (z wierszy 19-39 kodu) wyodrębnia wyrazy (jej rozwinięcie na następnej stronie).

Przykład 3 – wyodrębnienie wyrazów z pliku

```
19     while (p<temp.length())
20     {
21         //w pobranym lancuchu przeskakuje znaki
22         //spacja { } ( )
23         while(p<temp.length() && (temp[p]==' '
24             || temp[p]=='{' || temp[p]=='{'
25             || temp[p]=='(' || temp[p]==')')) {
26             p++;
27         } //dopuki nie trafie na jeden z pomianowanych znakow
28         //- pobieram znki i skladam n nich lancuch s
29         s="";
30         while (p<temp.length() && temp[p]!=' '
31             && temp[p]!='{' && temp[p]!='{'
32             && temp[p]!='(' && temp[p]!=')') {
33             s+=temp[p];
34             p++;
35         }
36         //wstawiam uczytany wyraz do tablicy wyrazow
37         wyrazy[lw]=s;
38         lw++;
39     }
```

- Ten fragment programu pracuje na pobranej z pliku linii – zapisanej w zmiennej temp.
- Jego zadaniem jest wyodrębnienie wyrazów i zapisanie ich w kolejnych polach tablicy wyrazy[]
- Wewnątrz pętli while (linia 19) pracują dwie kolejne. Pierwsza (linia 22-26) pomija spacje i nawiasy, druga (linia 29-34) wczytuje kolejne znaki (niebędące spacją ani nawiasem) i dopisuje do łańcucha s.
- Na koniec (linie 36-37) łańcuch s wstawiany jest do tablicy wyrazy[] a jej licznik zwiększany o 1.

Przykład 4 – Sito Erastotenesa

Zapisujemy kolejno wszystkie liczby w tabeli.

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Bierzemy pierwszą liczbę i począwszy od następnej wykreślamy wszystkie liczby, które są przez nią podzielne.

2 3 5 7 9 11 13 15 17 19

Bierzemy kolejną liczbę i począwszy od następnej wykreślamy liczby podzielne przez nią.

2 3 5 7 11 13 17 19

W tym momencie możemy zakończyć nasze poszukiwania. Algorytm "mówi", że kolejne wykreślenia należy powtarzać, nie dalej jak do liczby będącej zaokrąglonym w dół pierwiastkiem zakresu (naszym przypadku 4)

Przykład 4 – Sito Erastotenesa

```
1  #include<iostream>
2  #include<math.h>
3  using namespace std;
4  int main()
5  {
6      int liczba, i, j, pierwiastek, tab[1000];
7      cout << "Podaj maksimum: ";
8      cin >> liczba;
9      pierwiastek = (int) sqrt((double) liczba);
10     for (i = 0; i <= liczba; i++)
11         tab[i] = i;
12     for (i = 2; i <= pierwiastek; i++)
13     {
14         if (tab[i] != 0)
15             for (j = 2*i; j <= liczba; j+= i)
16                 tab[j] = 0;
17     }
18     cout << "Liczby pierwsze z przedzialu <2," << liczba << "> to:\n";
19     for (i = 2; i <= liczba; i++)
20         if (tab[i] != 0)
21             cout << tab[i] << " ";
22     cout << endl;
23     return 0;
24 }
```

Przykład 5 – Algorytm bisekcji

```
1  #include <iostream>
2  #include <cstdlib>
3  #include <ctime>
4
5  using namespace std;
6  void generowanie(long *tab);
7  int wyszukiwanie(long *tab, int co);
8
9  int main()
10 {
11     long tablica[1000];
12     srand(time(NULL));
13     generowanie(tablica);
14     cout << endl << endl << wyszukiwanie(tablica, 1500);
15     return 0;
16 }
17
18 void generowanie(long *tab)
19 {
32 int wyszukiwanie(long *tab, int co)
33 {
```

Przykład 5 – Algorytm bisekcji

```
18 void generowanie(long *tab)
19 {
20     long *p = tab;
21     *p=rand()%20;
22     p++;
23     for (int i=1; i<1000; i++)
24     {
25         *p=*(p-1)+rand()%20+1;
26         p++;
27     }
28     p=tab;
29     for (int i=1; i<1000; i++)
30     cout << *p++ << " ";
31 }
```

Generujemy 1000-elementową tablicę liczb całkowitych losowo rosnących.

Każda następną liczbą jest większa od poprzedniej o wartość z zakresu <1-20>

Przykład 5 – Algorytm bisekcji

```
32 int wyszukiwanie(long *tab,int co)
33 {
34     int pocz=0, kon=999,mid=500;
35     while(pocz!=kon)
36         if (*(tab+mid) ==co)
37             {
38                 return mid;
39             }
40     else if (*(tab+mid)>co)
41         {
42             kon=mid-1;
43             mid = (pocz+kon)/2;
44         }
45     else
46         {
47             pocz=mid+1;
48             mid = (pocz+kon)/2;
49         }
50     return pocz;
51 }
```

Wyszukiwanie kończymy, gdy znajdziemy w tablicy liczbę równą podanej, lub, gdy takiej liczby nie ma w tablicy – gdy krańce przedziału „pocz” i „kon” zetkną się.

To drugie rozwiązanie wskaże miejsce gdzie zadaną liczbę należałoby wstawić.

Algorytm można łatwo zmodyfikować, tak, aby podawał, czy liczbę znaleziono, czy nie.



Literatura:

W prezentacji wykorzystano przykłady i fragmenty:

- Grębosz J. : ***Symfonia C++***, ***Programowanie w języku C++ orientowane obiektowo***, Wydawnictwo Edition 2000.
- Jakubczyk K.: *Turbo Pascal i Borland C++ Przykłady*, Helion.

Warto zajrzeć także do:

- Sokół R. : ***Microsoft Visual Studio 2012 Programowanie w Ci C++***, Helion.
- Kernighan B.W., Ritchie D. M.: ***język ANSI C***, Wydawnictwo Naukowo Techniczne.

Dla bardziej zaawansowanych:

- Grębosz J. : ***Pasja C++***, Wydawnictwo Edition 2000.
- Meyers S.: ***język C++ bardziej efektywnie***, Wydawnictwo Naukowo Techniczne

