Lekcja 12. (p)

Temat: **Dodawanie ułamków, czyli jak wykorzystać NWW i NWD w programie komputerowym.**

Cele lekcji:

Zapoznanie z funkcjami w języku C++ i ich rolą w programie.
Wykorzystanie algorytmów NWW i NWD w praktyce.

Uczeń:

– układa i tłumaczy działanie, algorytmu dodawania ułamków zwykłych z wykorzystaniem algorytmu znajdowania NWW
– tworzy funkcje i używa ich w programie w C++
– dyskutuje strukturę ułożonego programu i uzasadnia użycie funkcji w kontekście rozbudowy programu

Przebieg lekcji:

1. **Stworzenie i analiza algorytmu wyszukującego dodającego ułamki zwykłe z wykorzystaniem NWD.**

2. **Utworzenie programu w C++ realizujące w/w zadanie:**
   a. **deklaracja zmiennej (typy zmiennych);**
   b. **pętla for i jej indeksowanie**
   c. **operatory warunkowe : if, else, if else, n%i**
   d. **testowanie program**

Zadanie domowe:

Powtórz zakres ćwiczenia z lekcji, przetestuj program.

Ćwiczenie z podręcznika – str. 96

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

int NWD(int a, int b)
int i;
while(b!=0)                 //obliczanie NWD
            {
               i = b;
               b = a%b;
               a = i;
            }
return a;
}

int main()
{
        int nww, a, b, i, l1, l2, m1, m2, licznik, mianownik;

        cout<<"Podaj licznik pierwszego ułamka  ";
        cin >> l1;
        cout << "Podaj mianownik pierwszego ułamka  ";
        cin >> m2;

        cout<<"Podaj licznik drugiego ułamka  ";
        cin >> l2;
        cout << "Podaj mianownik drugiego ułamka  ";
        cin >> m2;

        a=m2;
        b=m1;

        a=NWD(a,b);
```

```
            nww=m1*m2/a;

            mianownik = nww;
            licznik = mianownik/m1*l1 + mianownik/m2*l2;

            cout<<l1<<"/"<<m1<<" + "<<l2<<"/"<<m2<<" = "<<licznik<<"/"<<mianownik;

            return 0;
}
```

Rozwiązanie rozszerzające do lekcji 13-tej.

Sprawdzenie programu dodającego ułamki zwykłe z wykorzystaniem NWD i NWW.

```
#include<iostream>
#include<cstdlib>
using namespace std;
int A,a1,a2,B,b1,b2;
int c1,c2,d1,d2,m,mianownik;
int X,x1,x2,W,w1,w2;
int a,b;
int NWD(int a, int b)
{
   while(a!=b)
     if(a>b)
        a-=b;     //lub a = a - b;
     else
        b-=a;     //lub b = b-a
   return a;      // lub b - obie zmienne przechowują wynik NWD(a,b)
}

int NWW(int a, int b)
{
         int pom;
pom=(a*b)/NWD(a,b);
   return pom;
}

int main()
{
   cout <<"podaj część całkowitą liczby 1 (A)"; cin >> A;
   cout <<"podaj licznik liczby 1 (a1)"; cin >> a1;
   cout <<"podaj mianownik liczby 1 (b1)"; cin>> a2;

   cout <<"podaj część całkowitą liczby 2 (B)"; cin >> B;
   cout <<"podaj licznik liczby 2 (b1)"; cin >> b1;
   cout <<"podaj mianownik liczby 2 (b2)"; cin >> b2;
  cout <<endl<<"Wprowadziłeś ułamki"<<endl<<endl;

  cout <<"   "<<a1<<"         "<<b1<<endl;
  cout <<A<<" ------"<<" + "<<B<<" -------"<<endl;
  cout <<"   "<<a2<<"         "<<b2<<endl;

  cout <<"postac ogolna zapisu ułamków"<< endl;
  c1=A*a2+a1;
```

```
  c2=a2;
  d1=B*b2+b1;
  d2=b2;
   cout <<c1<<"         "<<d1<<endl;
   cout <<"------"<<" +  -------"<<endl;
   cout <<c2<<"         "<<d2<<endl;
m=NWD(c2,d2);
  cout << "wspólny dzielnik wynosi = "<<m<< endl;
mianownik=NWW(c2,d2);
  cout << "wspólny mianownik = "<<mianownik<< endl;
cout << "postac ułamka po wyznaczeniu wspólnegp mianownika"<<endl;
x1= mianownik/c2*c1+mianownik/d2*d1;
x2=mianownik;
cout <<x1<<endl;
cout <<"------"<<endl;
cout <<x2<<endl;
if (x1>x2)
{ W=x1/x2; w1=x1-W*x2; w2=x2;
cout <<"   "<<w1<<endl;
cout <<W<<" ------"<<endl;
cout <<"   "<<w2<<endl;
m=NWD(w1,w2);
cout <<"wspólny dzielnik dla licznika i mianownika = "<< m<<endl;

cout << "ułamek doprowadzony do najprostrzej postaci"<<endl;
w1=w1/m ;w2=w2/m;
cout <<"   "<<w1<<endl;
cout <<W<<" ------"<<endl;
cout <<"   "<<w2<<endl;
}
return 0;
}
```