

Wydział WIMiP	Imię i nazwisko: Zuzanna Będkowska	Rok: 2	Grupa: 1	Data: 24.04.2022
Metody Numeryczne	Temat: Całkowanie numeryczne za pomocą kwadratury Gaussa-Legendre'a			

**Zadanie: wyznaczenie wartości całek oznaczonych danych funkcji za pomocą kwadratury Gaussa-Legendre'a**

Zgodnie z instrukcją, w trakcie zajęć stworzono następujący kod:

```

1  #include <iostream>
2  #include <fstream>
3  #include <cmath>
4  #include <iomanip>
5  #include <vector>
6
7  using namespace std;
8
9  double sinus(double x)
10 {
11     return sin(x);
12 }
13
14 double parabola(double x)
15 {
16     return (pow(x, 2) + 2 * x + 5);
17 }
18
19 double expo(double x)
20 {
21     double e = 2.718281828459;
22     return (pow(e, x));
23 }
24
25 void cztery_wezly(double a, double b, double (f)(double))
26 {
27     vector<double> wagi;
28     double a1 = 1.0 / 36.0 * (18.0 - sqrt(30.0));
29     double a2 = 1.0 / 36.0 * (18.0 + sqrt(30.0));
30     wagi.push_back(a1);
31     wagi.push_back(a2);
32     wagi.push_back(a2);
33     wagi.push_back(a1);
34     vector<double> wezly;
35     double x1 = (-1.0) / 35.0 * sqrt(525.0 + 70.0 * sqrt(30.0));
36     double x2 = (-1.0) / 35.0 * sqrt(525.0 - 70.0 * sqrt(30.0));
37     wezly.push_back(x1);

```

```

38     wezly.push_back(x2);
39     wezly.push_back(-1.0*x2);
40     wezly.push_back(-1.0*x1);
41     vector<double> t(4, 0);
42     for (int i = 0; i < 4; ++i)
43     {
44         t[i] = (a + b) / 2.0 + ((b - a) / 2.0) * wezly[i];
45     }
46     double wynik = 0.0;
47     for (int i = 0; i < 4; ++i)
48     {
49         wynik += (b - a) / 2.0 * (wagi[i] * f(t[i]));
50     }
51     cout << "Wynik calkowania: " << wynik << "\n";
52 }
53
54 void dwa_wezly(double a, double b, double (f)(double))
55 {
56     vector<double> wagi(2, 1.0);
57     double x1 = (-1) * sqrt(3.0) / 3.0;
58     double x2 = sqrt(3.0) / 3.0;
59     double t1 = (a + b) / 2 + (b - a) / 2 * x1;
60     double t2 = (a + b) / 2 + (b - a) / 2 * x2;
61     double wynik = (b - a) / 2.0 * (wagi[0] * f(t1) + wagi[1] * f(t2));
62     cout << "Wynik calkowania: " << wynik << "\n";
63 }
64
65 int main()
66 {
67     cout << "Calkuje funkcje: sin(x) w przedziale 0,5 - 2,5\n\t-Kwadratura dwuwzlowa: ";
68     dwa_wezly(0.5, 2.5, sinus);
69     cout << "\t-Kwadratura czterowzlowa: ";
70     cztery_wezly(0.5, 2.5, sinus);
71     cout << "Calkuje funkcje: x^2 + 2x + 5 w przedziale 0,5 - 5,0\n\t-Kwadratura dwuwzlowa: ";
72     dwa_wezly(0.5, 5.0, parabola);
73     cout << "\t-Kwadratura czterowzlowa: ";
74     cztery_wezly(0.5, 5.0, parabola);
75
76     cout << "Calkuje funkcje: exp(x) w przedziale 0,5 - 5,0\n\t-Kwadratura dwuwzlowa: ";
77     dwa_wezly(0.5, 5.0, expo);
78     cout << "\t-Kwadratura czterowzlowa: ";
79     cztery_wezly(0.5, 5.0, expo);
80
81 }

```

Program wyznacza wartości całek za pomocą kwadratury dwuwęzłowej i czterowęzłowej. Odpowiednie funkcje wyznaczają węzły kwadratury i ich wagi i tworzy wielomian interpolacyjny z iloczynów wagi i wartości funkcji w danym węźle. Efekt działania jest następujący:

```

Konsola debugowania programu Microsoft Visual Studio
Calkuje funkcje: sin(x) w przedziale 0,5 - 2,5
-Kwadratura dwuwzlowa: Wynik calkowania: 1.67163
-Kwadratura czterowzlowa: Wynik calkowania: 1.67873
Calkuje funkcje: x^2 + 2x + 5 w przedziale 0,5 - 5,0
-Kwadratura dwuwzlowa: Wynik calkowania: 88.875
-Kwadratura czterowzlowa: Wynik calkowania: 88.875
Calkuje funkcje: exp(x) w przedziale 0,5 - 5,0
-Kwadratura dwuwzlowa: Wynik calkowania: 138.621
-Kwadratura czterowzlowa: Wynik calkowania: 146.757

C:\Users\Zuza\source\repos\MetodyNumeryczne_lab7\x64\Debug\MetodyNumeryczne_lab7.exe (proces 22112) zakończono z kodem 0
Naciśnij dowolny klawisz, aby zamknąć to okno...

```