Условие:

Выполнить программную реализацию вычисления определенного интеграла по формулам Ньютона-Котеса для функции, заданной таблично в пяти узлах:

|  |  |
| --- | --- |
| x | y |
| 0,281 | 1,90985 |
| 0,283 | 1,91867 |
| 0,285 | 1,92752 |
| 0,287 | 1,93642 |
| 0,289 | 1,94421 |

Исходный код программы:  
  
#include **"stdafx.h"**#include **<iostream>**#include **<cmath>  
using namespace** std;  
  
**double** F(**double** x);  
**double** Nuton\_Kotens(**double** ba, **int** n, **double** a, **double** h);  
  
**int** main()  
{  
 setlocale(LC\_ALL, **"RUS"**);  
 **int** n;  
 **double** a, b, h;  
 cout << **"Интервал от -> "**;  
 cin >> a;  
 cout << **"\nИнтервал от "** << a << **" до -> "**;  
 cin >> b;  
 cout << **"Шагов "**;  
 cin >> n;  
  
 h = (b - a) / n;  
  
 **if** (n <= 7)  
 cout << **"\nОтвет: "** << Nuton\_Kotens(b - a, n, a, h);  
 **else** cout << **"число шагов больше 6!"**;  
 cout << **"\n"**;  
 system(**"pause"**);  
 **return** 0;  
}  
  
**double** F(**double** x) {  
 **return** 1 / (x + 2);  
}  
  
**double** Nuton\_Kotens(**double** ba, **int** n, **double** a, **double** h)  
{  
 **double** C[7], s = 0;  
 **switch** (n) {  
 **case** 1: {  
 C[0] = (ba / 2);  
 C[1] = C[0];  
 **break**;  
 }  
 **case** 2: {  
 C[0] = (ba / 6);  
 C[1] = (4 \* ba / 6);  
 C[2] = C[0];  
 **break**;  
 }  
 **case** 3: {  
 C[0] = (ba / 8);  
 C[1] = (3 \* ba / 8);  
 C[2] = C[1];  
 C[3] = C[0];  
 **break**;  
 }  
 **case** 4: {  
 C[0] = (7 \* ba / 90);  
 C[1] = (16 \* ba / 45);  
 C[2] = (2 \* ba / 15);  
 C[3] = C[1];  
 C[4] = C[0];  
 **break**;  
 }  
 **case** 5: {  
 C[0] = (19 \* ba / 288);  
 C[1] = (25 \* ba / 96);  
 C[2] = (25 \* ba / 144);  
 C[3] = C[2];  
 C[4] = C[1];  
 C[5] = C[0];  
 **break**;  
 }  
 **case** 6: {  
 C[0] = (41 \* ba / 840);  
 C[1] = (9 \* ba / 35);  
 C[2] = (9 \* ba / 280);  
 C[3] = (34 \* ba / 105);  
 C[4] = C[2];  
 C[5] = C[1];  
 C[6] = C[0];  
 **break**;  
 }  
 };  
 **for** (**int** i = 0; i <= n; i++)  
 s += F(a + i \* h) \* C[i];  
 **return** s;  
}

Скриншоты программы:



