4.2 - Labs (2)

* **Lab 4.1.9 (1) - Multi-dimensional arrays of variable length [B]**

#include <iostream>

#include<cstdlib>

#include<ctime>

using namespace std;

int main()

{

float n, cu, gr, sum = 0, sumf = 0;

double fin;

cout << "How much Courses? ";

cin >> n;

cout << endl;

if (n <= 5)

{

for (int i = 0; i < n; i++)

{

cout << "Courses " << i + 1 << " grades: ";

cin >> cu;

if (cu <= 10)

{

cout << "Enter " << cu << " marks:" << endl;

for (int j = 0; j < cu; j++)

{

cin >> gr;

if (gr <= 5)

sum += gr;

else

{

cout << "Error, try again" << endl;

break;

}

}

fin = sum / cu;

cout << "Final: " << fin << endl;

sum = 0;

sumf += fin;

}

else

{

cout << "Error, try again" << endl;

break;

}

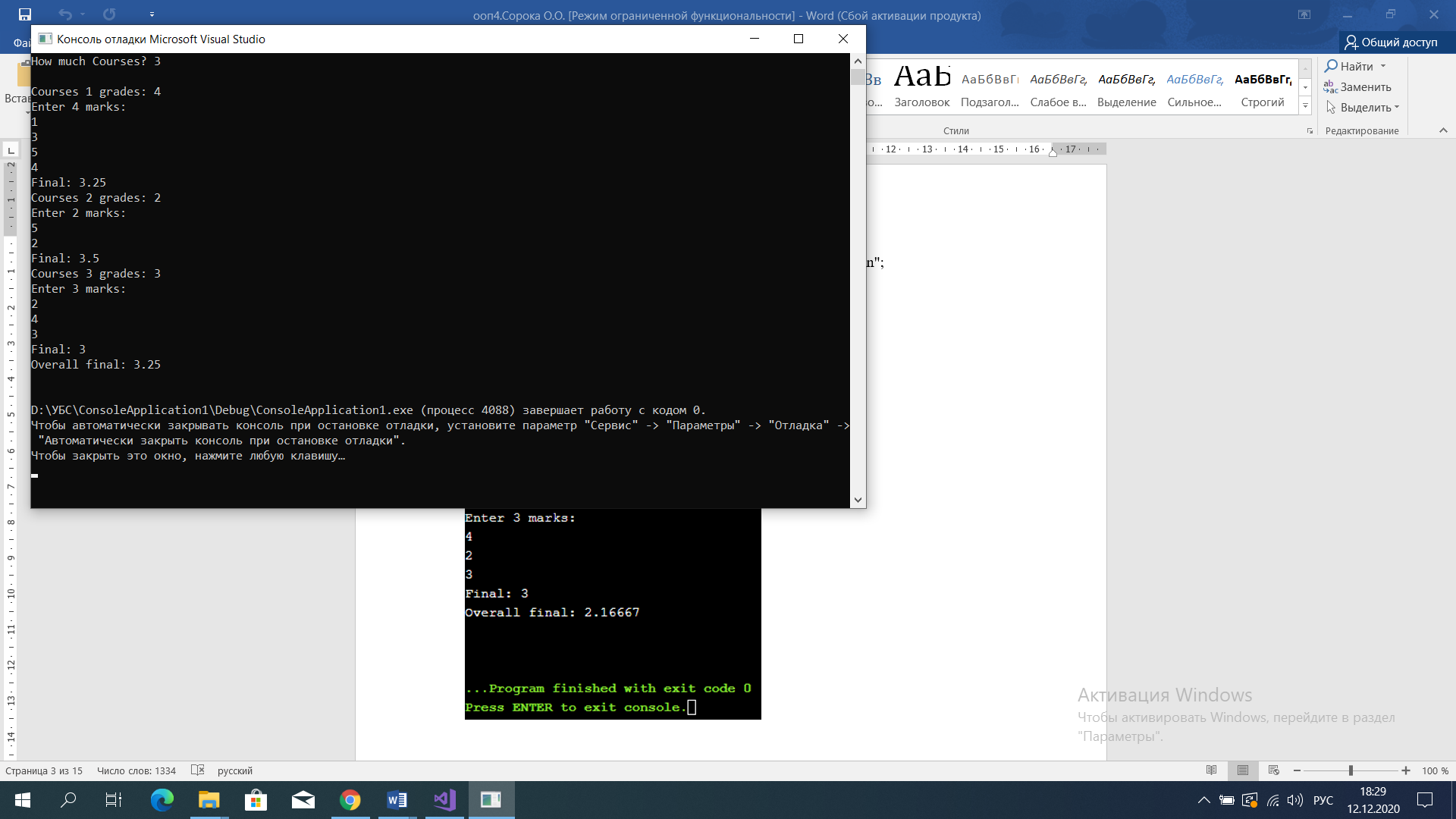
}

cout << "Overall final: " << sumf / n << "\n\n";

}

return 0;

}



* **Lab 4.2.5 (1) Range of integer types [A]**

#include <iostream>

#include<cstdlib>

#include<ctime>

using namespace std;

int main()

{

long int n, summ = 0;

cout << "Enter n: ";

cin >> n;

if (n >= 1 && n <= 1000000)

{

for (int i = 1; i < n + 1; i++)

summ = summ + i;

cout << summ << endl;

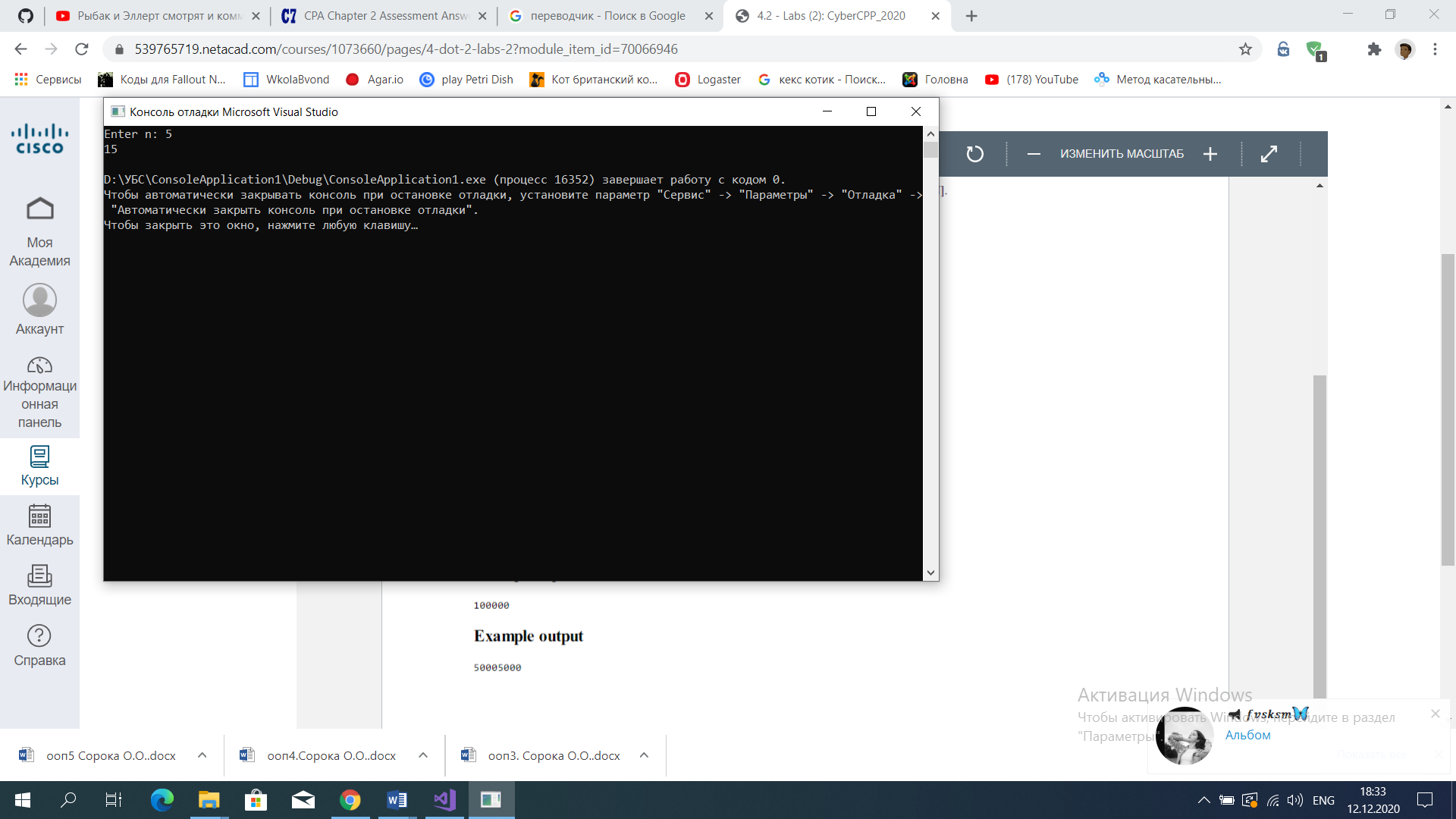
}

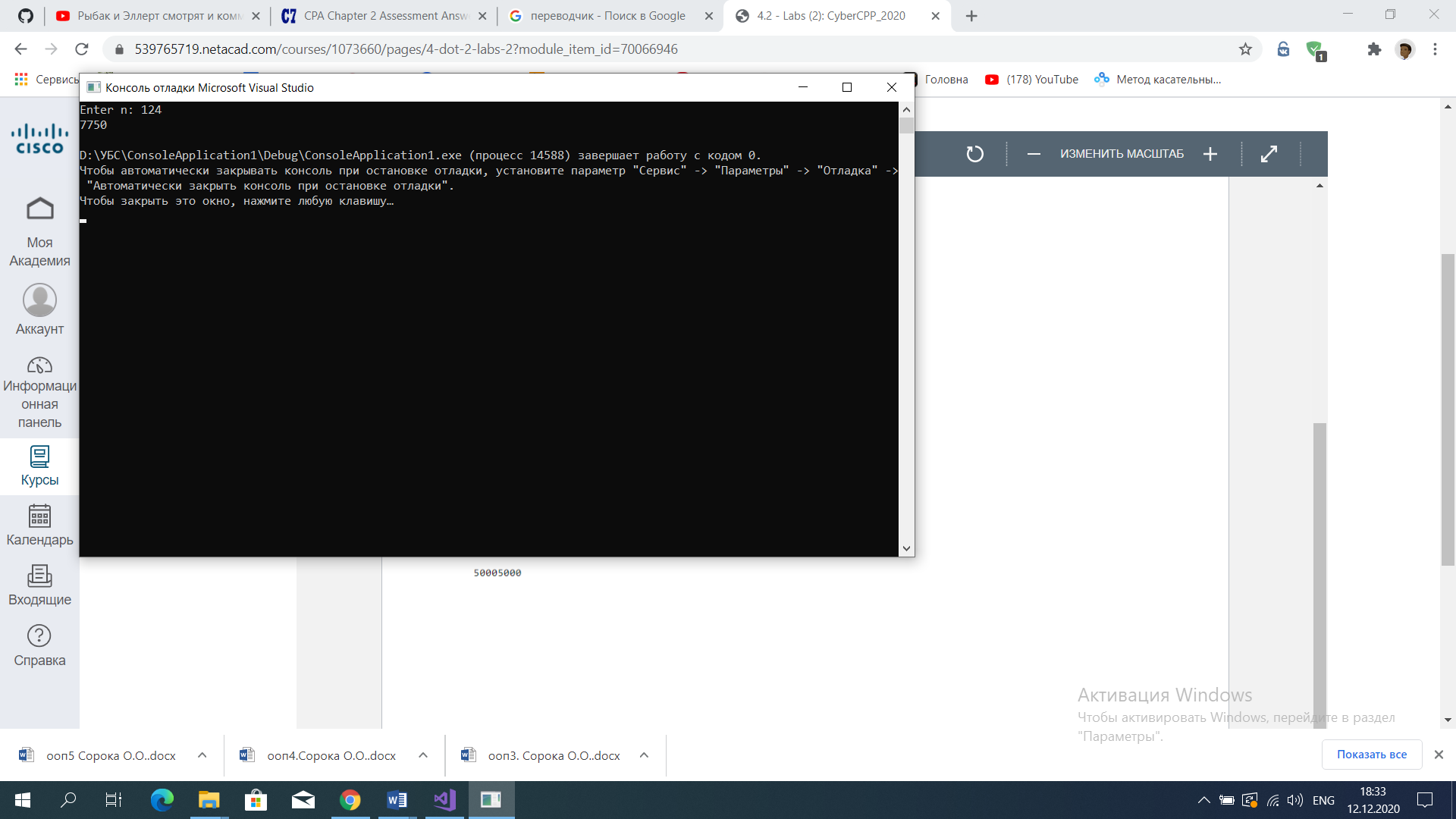
else

cout << "Error" << endl;

return 0;

}





* **Lab 4.2.7 (1) Comparing floating-point numbers [A]**

#include <iostream>

#include<cstdlib>

#include<ctime>

using namespace std;

bool is\_close(double a, double b, double tolerance)

{

if (b >= a)

{

double k = b - a;

if (k <= tolerance)

return true;

else

return false;

}

else

{

double k = a - b;

if (k <= tolerance)

return true;

else

return false;

}

return false;

}

int main()

{

if (0.3 == 3 \* 0.1)

cout << "The numbers are equal";

else

cout << "The numbers are not equal";

cout << endl;

if (is\_close(0.3, 3 \* 0.1, 0.00000001))

cout << "The numbers are close enough";

else

cout << "The numbers are not close enough";

cout << endl;

if (is\_close(3 \* 0.1, 0.3, 0.00000001))

cout << "The numbers are still close enough";

else

cout << "The numbers are not close enough";

cout << endl;

if (is\_close(3 \* 0.1, 0.31, 0.00000001))

cout << "The numbers are still close enough";

else

cout << "The numbers are not close enough";

cout << endl;

return 0;

}

