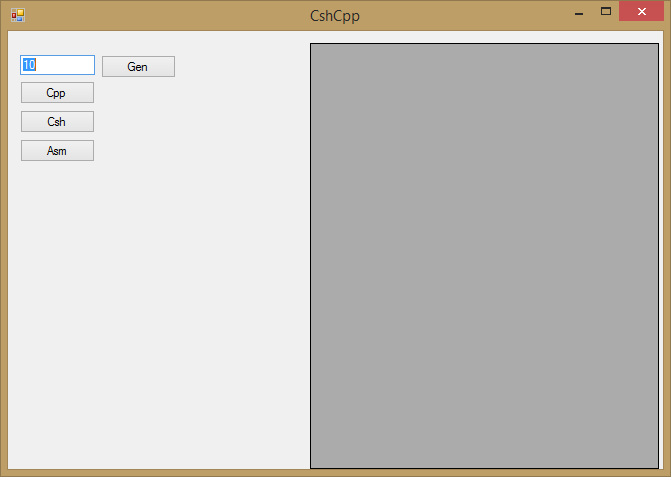
Отчет по программе CshCpp

**Модель:** Нахождение суммы чисел

**Описание интерфейса:**



Кнопка «Gen» генерирует 2 массива произвольных чисел и выводит в DataGridView; «Csh» - суммирует поэлементно созданные массивы в коде C#; «Cpp»- суммирует поэлементно созданные массивы в коде динамической библиотеки C++ ; «Asm» - суммирует элементы массива; результаты суммирования выводятся в DataGridView; а время выполнения в Lebel.

**Описание программы:**

Класс CshCppGSA

CshCppWrp ap = new CshCppWrp();

CshCppWprAsm apasm = new CshCppWprAsm();

private float[] A;

private float[] B;

private float[] C;

public void Gen (int \_l)

{

A = new float[\_l + 4 - (\_l % 4)];

B = new float[\_l + 4 - (\_l % 4)];

C = new float[\_l + 4 - (\_l % 4)];

int i;

Random rnd = new Random();

for (i = 0; i < \_l; i++)

{

A[i] = rnd.Next(0, 10);

B[i] = rnd.Next(0, 10);

}

for(i=\_l; i < \_l + 4 - (\_l % 4); i++)

{

A[i] = 0;

B[i] = 0;

}

}

public void Nu (DataGridView \_dgC)

{

int i;

for (i = 0; i < C.Length; i++)

\_dgC.Rows[i].Cells[2].Value = null;

}

public void Get (DataGridView \_dgC)

{

int i;

\_dgC.RowCount = C.Length;

\_dgC.ColumnCount = 3;

for (i = 0; i < C.Length; i++)

{

\_dgC.Rows[i].Cells[0].Value = A[i].ToString();

\_dgC.Rows[i].Cells[1].Value = B[i].ToString();

\_dgC.Rows[i].Cells[2].Value = C[i].ToString();

}

}

public string SumCsh()

{

if ((A == null) || (B == null))

{

return @"No array";

}

try

{

int i;

Stopwatch timer = new Stopwatch();

timer.Restart();

for (i = 0; i < C.Length; i++)

C[i] = A[i] + B[i];

timer.Stop();

return string.Format(@"Timer:{0}", timer.Elapsed);

}

catch (Exception ex)

{

return ex.Message;

}

}

public unsafe string SumCppDll ()

{

float \*m;

if ((A == null) || (B == null))

{

return @"Arrey no";

}

try

{

Stopwatch timer = new Stopwatch();

timer.Restart();

m = ap.Sum(A, B, C);

timer.Stop();

return string.Format(@" Timer:{0}", timer.Elapsed);

}

catch (Exception ex)

{

return ex.Message;

}

}

public unsafe string SumAsmDll()

{

float l;

if ((A == null) || (B == null))

{

return @"Arrey no";

}

try

{

Stopwatch timer = new Stopwatch();

timer.Restart();

l = apasm.SumAsm(A);

timer.Stop();

return string.Format(@" Timer:{0} Res:{1}", timer.Elapsed, l);

}

catch (Exception ex)

{

return ex.Message;

}

}

Класс CshCppWrp

public unsafe float\* Sum (float[] ArrA, float[] ArrB, float[] ArrC)

{

return ArrSum(ArrA, ArrB, ArrC, ArrA.Length);

}

[DllImport("CppAP.dll", EntryPoint = "Sum", CharSet = CharSet.Auto, SetLastError = false)]

private unsafe static extern float\* ArrSum(float[] arreyA, float[] arreyB, float[] arreyC, int length);

Класс CshCppWrpAsm

public unsafe float SumAsm (float[] arrayA)

{

return ArrSumExx(arrayA, arrayA.Length);

}

[DllImport("CppAP.dll", EntryPoint = "SumX", CharSet = CharSet.Auto, SetLastError = false)]

private unsafe static extern float ArrSumExx(float[] arrA, int length);

dll DllSum

#include <cstdint>

#ifdef \_\_cplusplus

extern "C" {

#endif

\_\_declspec(dllexport) float\* Sum(float\* ArrayA, float\* ArrayB, float\* ArrayC, int length) {

int i;

for (i = 0; i < length; i++) {

ArrayC[i]= ArrayA[i] + ArrayB[i];

}

return ArrayC;

}

\_\_declspec(dllexport) float SumX(float\* Array, int length) {

float result = 0;

float Res[4];

int len = length \* 4;

float\* p;

p = Array;

\_asm

{

push ECX

push EDI

push ESI

mov EDI, len

mov ECX, 0

mov ESI, p

pxor xmm0, xmm0 //| 0 | 0 | 0 | 0 |

Back :

movups xmm1, [ESI][ECX] //Array[ECX]

addps xmm0, xmm1

add ECX, 16

cmp ECX, EDI

jne Back

movups Res, xmm0

pop EDI

pop ECX

}

result = Res[0] + Res[1] + Res[2] + Res[3];

return result;//len;

}

#ifdef \_\_cplusplus

}

#endif

**Протокол отладки:**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Исходные данные | Результат | Ошибки |
| 1 |  |  | Нет ошибок |
| 2 |  |  | Нет ошибок |
| 3 |  |  | Нет ошибок |
| 4 |  |  | Нет ошибок |