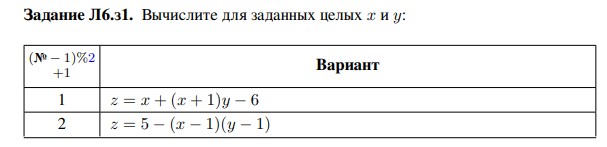
****

**#include <cstdio>**

**#include <iostream>**

**using namespace std;**

**int main() {**

**int x = 2, y = 3, z;**

**asm (**

**"movl %[X], %%eax\n\t"**

**"subl $1, %%eax\n\t"**

**"movl %[Y], %%ebx\n\t"**

**"subl $1, %%ebx\n\t"**

**"imull %%ebx\n\t"**

**"imull $-1, %%eax\n\t"**

**"addl $5, %%eax\n\t"**

**"movl %%eax, %0"**

**:"=m"(z)**

**:[X]"m"(x), [Y]"m"(y)**

**:"cc", "eax", "ebx", "memory"**

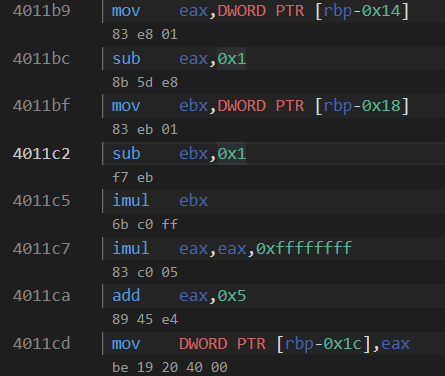
**);**

**cout << "z = " << z << endl;**

**return 0;**

**}**

**z = 3**

****

****

****

**#include <cstdio>**

**#include <iostream>**

**using namespace std;**

**int main() {**

**cout << "Ассемблер: " << endl;**

**int x = 5, z;**

**asm (**

**"lea 0xFFFFFF9F(,%%eax,8), %%eax\n\t"**

**:"=a"(z)**

**:[X]"a"(x)**

**);**

**cout << "z = " << z << endl;**

**return 0;**

**}**

**Ассемблер:**

**z = -57**

****

**#include <cstdio>**

**#include <iostream>**

**using namespace std;**

**int main() {**

**unsigned int C;**

**asm(**

**"cpuid\n"**

**: "=c"(C)**

**: "a"(1)**

**: "ebx", "edx");**

**bool AVX\_bit = (C & (1 << 28)) != 0;**

**cout << AVX\_bit << endl;**

**unsigned int D;**

**asm(**

**"cpuid\n"**

**: "=d"(D)**

**: "a"(1)**

**: "ebx", "ecx"**

**);**

**bool SSE\_bit = (D & (1 << 25)) != 0;**

**cout << SSE\_bit << endl;**

**unsigned int D0;**

**asm(**

**"cpuid\n"**

**: "=d"(D0)**

**: "a"(1)**

**: "ebx", "ecx"**

**);**

**bool FPU\_bit = (D0 & (1 << 0)) != 0;**

**cout << FPU\_bit << endl;**

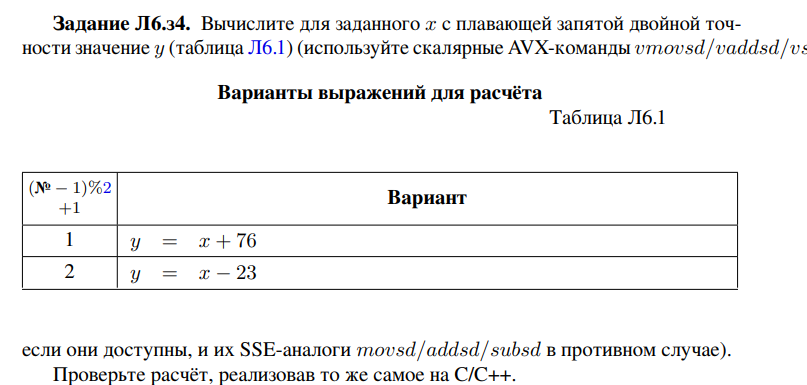
**return 0;**

**}**

**1**

**1**

**1**

****

**#include <cstdio>**

**#include <iostream>**

**using namespace std;**

**int main() {**

**cout << "Ассемблер: " << endl;**

**double x = 14.6, y;**

**asm (**

**"vsubsd %[Sub], %[X], %[X]\n\t"**

**:"=x"(y)**

**:[X]"x"(x), [Sub]"x"(23.0)**

**:"cc"**

**);**

**cout << "y = " << y << endl;**

**cout << endl;**

**cout << "C++" << endl;**

**cout << "y = " << (x-23.0);**

**}**

**Ассемблер:**

**y = -8.4**

**C++**

**y = -8.4**

****

**#include <cstdio>**

**#include <iostream>**

**using namespace std;**

**int main() {**

**const double a = 23;**

**double x = 1, y;**

**asm(**

**"fldl %[X]\n\t" // st(0) = %[X]**

**"fsubl %[A]\n\t" // st(0) = st(0) - %[A]**

**"fstpl %[Y]\n\t" // %[Y] = st(0), стек пуст**

**:[Y]"=m"(y)**

**:[X]"m"(x), [A]"m"(a)**

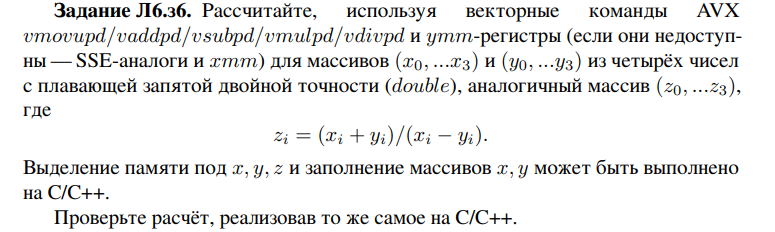
**:"cc"**

**);**

**cout << y;**

**return 0;**

**}**

**-22  
  
  
**

**#include <cstdio>**

**#include <iostream>**

**#include <ctime>**

**using namespace std;**

**const int N = 4;**

**void print\_mas(double\* mas)**

**{**

**for (int i = 0; i < N; i++)**

**cout << mas[i] << " ";**

**cout << endl;**

**}**

**int main() {**

**cout << "Ассемблер: " << endl;**

**double x[4], y[4], z[4];**

**for (int i = 0; i < N; i++)**

**{**

**x[i] = rand() % N + 1;**

**y[i] = rand() % N + 5;**

**}**

**print\_mas(x);**

**print\_mas(y);**

**asm (**

**"vmovupd %[X], %%ymm1\n\t"**

**"vmovupd %[Y], %%ymm2\n\t"**

**"vaddpd %%ymm1, %%ymm2, %%ymm3\n\t"**

**"vsubpd %%ymm2, %%ymm1, %%ymm4\n\t"**

**"vdivpd %%ymm4, %%ymm3, %%ymm1\n\t"**

**"vmovupd %%ymm1, %[Z]"**

**:[Z]"=m"(z)**

**:[X]"m"(x), [Y]"m"(y)**

**:"memory", "ymm1", "ymm2", "ymm3", "ymm4"**

**);**

**print\_mas(z);**

**cout << endl;**

**cout << "С++: " << endl;**

**print\_mas(x);**

**print\_mas(y);**

**for (int i = 0; i < N; i++)**

**z[i] = (x[i]+y[i])/(x[i]-y[i]);**

**print\_mas(z);**

**}**

**Ассемблер:**

**4 2 2 3**

**7 8 8 5**

**-3.66667 -1.66667 -1.66667 -4**

**С++:**

**4 2 2 3**

**7 8 8 5**

**-3.66667 -1.66667 -1.66667 -4**

****

**#include <cstdio>**

**#include <iostream>**

**#include <ctime>**

**using namespace std;**

**const int N = 4;**

**void print\_mas(double\* mas)**

**{**

**for (int i = 0; i < N; i++)**

**cout << mas[i] << " ";**

**cout << endl;**

**}**

**int main() {**

**cout << "Ассемблер: " << endl;**

**alignas(32) double x[4], y[4], z[4];**

**for (int i = 0; i < N; i++)**

**{**

**x[i] = rand() % N + 1;**

**y[i] = rand() % N + 5;**

**}**

**print\_mas(x);**

**print\_mas(y);**

**asm (**

**"vmovapd %[X], %%ymm1\n\t"**

**"vmovapd %[Y], %%ymm2\n\t"**

**"vaddpd %%ymm1, %%ymm2, %%ymm3\n\t"**

**"vsubpd %%ymm2, %%ymm1, %%ymm4\n\t"**

**"vdivpd %%ymm4, %%ymm3, %%ymm1\n\t"**

**"vmovapd %%ymm1, %[Z]"**

**:[Z]"=m"(z)**

**:[X]"m"(x), [Y]"m"(y)**

**:"memory", "ymm1", "ymm2", "ymm3", "ymm4"**

**);**

**print\_mas(z);**

**}**

**Ассемблер:**

**4 2 2 3**

**7 8 8 5**

**-3.66667 -1.66667 -1.66667 -4**