**Лабораторная работа №2.**

**Ветютнев Даниил**

**Шульга Семен**

**Задание 1.**Разработайте программу на языке C++, вычисляющую три целых выражения от целого аргумента (в соответствии с вариантом).

**Задание 2.**Запустите программу и, используя инструменты отладчика (в частности, дизассемблер), изучите ассемблерный код, соответствующий вычислениям.

1

int main()

{

int x = 5;

int y = x \* 5;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int x = 5;

00CA1728 mov dword ptr [x],5

int y = x \* 5;

00CA172F imul eax,dword ptr [x],5

00CA1733 mov dword ptr [y],eax

2

int main()

{

int x = 9;

int y = x / 3;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int x = 9;

007C1728 mov dword ptr [x],9

int y = x / 3;

007C172F mov eax,dword ptr [x]

007C1732 cdq

007C1733 mov ecx,3

007C1738 idiv eax,ecx

007C173A mov dword ptr [y],eax

}

3

int main()

{

int x = -5;

if (x >= 0)

int y = 13;

else

int y = 7;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int x = 5;

00651DA8 mov dword ptr [x],5

if (x >= 0)

00651DAF cmp dword ptr [x],0 //сравнение с 0, не сохраняя

00651DB3 jl main+3Eh (0651DBEh)

int y = 13;

00651DB5 mov dword ptr [ebp-14h],0Dh

00651DBC jmp main+45h (0651DC5h)

else

int y = 7;

00651DBE mov dword ptr [ebp-20h],7

#3

**Задание 3.**Внесите в программу из задания 1, а) изменения сделайте переменные глобальными;

– измените тип с int на char, short, long и long long;

– измените тип с int на long double

|  |  |  |
| --- | --- | --- |
| Global1 | x = 5;  002A1DA8 mov dword ptr [x (02AA17Ch)],5  y = x \* 5;  002A1DB2 imul eax,dword ptr [x (02AA17Ch)],5  002A1DB9 mov dword ptr [y (02AA180h)],eax  } |  |
| Global2 | x = 9;  00FD1DA8 mov dword ptr [x (0FDA17Ch)],9  y = x / 3;  00FD1DB2 mov eax,dword ptr [x (0FDA17Ch)]  00FD1DB7 cdq  00FD1DB8 mov ecx,3  00FD1DBD idiv eax,ecx  00FD1DBF mov dword ptr [y (0FDA180h)],eax  } |  |
| Global3 | x = -5;  00171DA8 mov dword ptr [x (017A17Ch)],0FFFFFFFBh  if (x >= 0)  00171DB2 cmp dword ptr [x (017A17Ch)],0  00171DB9 jl main+47h (0171DC7h)  y = 13;  00171DBB mov dword ptr [y (017A180h)],0Dh  00171DC5 jmp main+51h (0171DD1h)  else  y = 7;  00171DC7 mov dword ptr [y (017A180h)],7  } |  |
| Short1 | short x = 5;  00051DA8 mov eax,5  00051DAD mov word ptr [x],ax  short y = x \* 5;  00051DB1 movsx eax,word ptr [x]  00051DB5 imul ecx,eax,5  00051DB8 mov word ptr [y],cx  } | Требуется копирование,  Работает не по адресу,  Результат не нужно копировать |
| Long1 | long x = 5;  009B1DA8 mov dword ptr [x],5  long y = x \* 5;  009B1DAF imul eax,dword ptr [x],5  009B1DB3 mov dword ptr [y],eax  } | Dword -32 bit, word – 16 bit |
| Longlong1 | long long x = 5;  00DB1DA8 mov dword ptr [x],5  00DB1DAF mov dword ptr [ebp-8],0  long long y = x \* 5;  00DB1DB6 push 0  00DB1DB8 push 5  00DB1DBA mov eax,dword ptr [ebp-8]  00DB1DBD push eax  00DB1DBE mov ecx,dword ptr [x]  00DB1DC1 push ecx  00DB1DC2 call \_\_allmul (0DB1375h)  00DB1DC7 mov dword ptr [y],eax  00DB1DCA mov dword ptr [ebp-18h],edx  } | Берет большие резделения между элементами |
| Longdouble1 | long double x = 5;  003F1DA8 movsd xmm0,mmword ptr [\_\_real@4014000000000000 (03F7BD0h)]  003F1DB0 movsd mmword ptr [x],xmm0  long double y = x \* 5;  003F1DB5 movsd xmm0,mmword ptr [x]  003F1DBA mulsd xmm0,mmword ptr [\_\_real@4014000000000000 (03F7BD0h)]  003F1DC2 movsd mmword ptr [y],xmm0  } | Sd- копирование двойных слов, |

**Задание 4.**

Оформите вычисления из задания 1, а) как целую функцию от целого аргумента. Опишите в отчёте код вызова функции. Как передаётся аргумент? Как возвращается значение?

int multiply(int x)

{

return x \* 5;

}

int main()

{

int x = 5;

int y = multiply(x);

return 0;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dissasembly

int y = multiply(x);

006A177F mov eax,dword ptr [x]

006A1782 push eax

006A1783 call multiply (06A10AAh)

006A1788 add esp,4

006A178B mov dword ptr [y],eax

**Задание 5.**

Измените тип аргумента и результата на вещественный. Опишите в отчёте код вызова функции. Как передаётся аргумент? Как возвращается значение?

float multiply(float x)

{

return x \* 5;

}

int main()

{

float x = 5;

float y = multiply(x);

return 0;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dissasembly

float y = multiply(x);

00F11785 push ecx

00F11786 movss xmm0,dword ptr [x]

00F1178B movss dword ptr [esp],xmm0

00F11790 call multiply (0F1137Fh)

00F11795 add esp,4

00F11798 fstp dword ptr [y]

**Задание 6.**

Используйте в функции статическую переменную. Как выглядит обращение к ней?

float multiply(static float x)

{

return x \* 5;

}

int main()

{

static float x = 5;

float y = multiply(x);

return 0;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dissasembly

float y = multiply(x);

006D1778 push ecx

006D1779 movss xmm0,dword ptr [x (06DA028h)]

006D1781 movss dword ptr [esp],xmm0

006D1786 call multiply (06D137Fh)

006D178B add esp,4

006D178E fstp dword ptr [y]