

Лабораторная работа 6 (0110 = 6)

Модули и функции. Описание функций на ассемблере

Операционная система ОС Windows, 64-разрядная ОС, соглашение о вызовах Microsoft x64 (x86-64).

Задание Л6.з1. (Вариант 1: $z(x, y) = -113 + x + 2y$)

Разработайте ассемблерную функцию, вычисляющую целое выражение от двух целых аргументов (таблица Л6.1), а также головную программу на языке C/C++, использующую разработанную функцию.

Бонус +1 балл, если вычисление производится одной командой *leq*. Для проверки реализуйте вычисление того же выражения на C/C++

Программный код:

C:

```
#include <stdio.h>
```

```
extern "C" {
```

```
int task1(int a, int b);
```

```
}
```

```
int main() {
```

```
    int x = 16;
```

```
    int y = -5;
```

```
    printf("Результат выражения в C++: %d\n", -113 + x + 2 * y);
```

```
    printf("Результат функции в ассемблере: %d\n", task1(x, y));
```

```
    return 0;
```

```
}
```

Ассемблер:

.data

.text

.globl task1

task1:

sub \$56, %rsp

mov %ecx, %edi

// получение входного параметра x

mov %edx, %esi

// получение входного параметра y

// Базовое смещение + (Масштабируемый индекс * Масштабный коэффициент) + Смещение

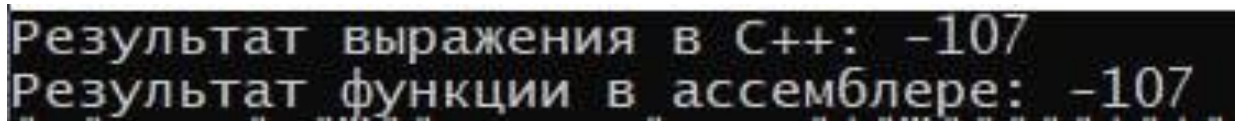
// edi + (esi * 2) - 113

lea -113(%edi, %esi, 2), %eax

add \$56, %rsp

retq

Вывод:



```
Результат выражения в C++: -107
Результат функции в ассемблере: -107
```

Задание Л6.з2. (Вариант 1: $z(x, y) = x - y$)

Разработайте ассемблерную функцию, вычисляющую выражение от двух чисел с плавающей запятой двойной точности x и y (таблица Л6.2), используя команды AVX *vsubsd* и *vdivsd* или их SSE-аналоги *subsd* и *divsd*, а также головную программу на языке C/C++.

Программный код:

C:

```
#include <stdio.h>
```

```
extern "C" {
```

```
double task2(double a, double b);
```

```
}
```

```
int main() {
```

```
    double x = 16.54;
```

```
    double y = -5.55;
```

```
    printf("Результат выражения в C++: %lf\n", x - y);
```

```
    printf("Результат функции в ассемблере: %lf\n", task2(x, y));
```

```
    return 0;
```

```
}
```

Ассемблер:

```
.data
```

```
.text
```

```
.globl task2
```

```
task2:
```

```
    sub $56, %rsp
```

```
    subsd %xmm1, %xmm0
```

```
    add $56, %rsp
```

```
retq
```

Вывод:

```
Результат выражения в C++: 22.090000
Результат функции в ассемблере: 22.090000
```

Задание Л6.з3.

Опишите на произвольном языке высокого уровня (включая C/C++) функцию с семью целочисленными параметрами, которая выводит свои параметры на экран и возвращает результат, равный второму параметру.

Разработайте головную программу на ассемблере, вызывающую эту функцию.

Программный код:

C:

```
#include <stdio.h>
```

```
extern "C"
```

```
int task3(int a, int b, int c, int d, int e, int f, int g) {
```

```
    printf("a = %d, b = %d, c = %d, d = %d, e = %d, f = %d, g = %d\n", a, b, c, d, e, f, g);
```

```
    return b;
```

```
}
```

Ассемблер:

```
.data
```

```
int_format_out: .string "Возвращаемое значение b из ассемблера: %d\n"
```

```
.text
```

```
.globl main
```

```
main:
```

```
    sub $24, %rsp
```

```
    mov $1, %ecx
```

```
    mov $2, %edx
```

```
    mov $3, %r8
```

```
    mov $4, %r9
```

```
mov $5, (%rsp)
```

```
mov $6, 8(%rsp)
```

```
mov $7, 16(%rsp)
```

```
sub $32, %rsp
```

```
call task3
```

```
add $32, %rsp
```

```
lea int_format_out(%rip), %rcx
```

```
mov %rax, %rdx
```

```
sub $32, %rsp
```

```
call printf
```

```
add $32, %rsp
```

```
add $24, %rsp
```

```
xor %eax, %eax
```

```
retq
```

Вывод:

```
a = 1, b = 2, c = 3, d = 4, e = 5, f = 6, g = 7
Возвращаемое значение b из ассемблера: 2
```