

Лабораторная работа №7

Изучение процесса разработки подпрограмм на языке ассемблера

1 Цель работы

1.1 Изучить процесс разработки подпрограмм с использованием ассемблера NASM.

1.2 Изучить процесс разработки программ на ассемблере состоящих из нескольких отдельных модулей.

2 Литература

2.1 Вызов функции ассемблера из C/C++. Ассемблер NASM. – Текст : электронный // METANIT, 2025. – URL: <https://metanit.com/assembler/nasm/7.2.php>

2.2 Функции. Ассемблер NASM. – Текст : электронный // METANIT, 2025. – URL: <https://metanit.com/assembler/nasm/4.1.php>

2.3 SASM. Simple crossplatform IDE for NASM, MASM, GAS, FASM assembly languages – Текст : электронный // SASM, 2025. – URL: <https://dman95.github.io/SASM/>

3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

4 Основное оборудование

4.1 Персональный компьютер.

5 Задание

При разработке функций требуется сохранять и восстанавливать значения регистров, используемых внутри функции.

Все разработанные функции необходимо протестировать.

5.1 Разработать подпрограмму на языке ассемблера, вычисляющую значение 2^x (x – неотрицательное). Для передачи данных использовать регистры. Результат нужно возвращать в вызывающий модуль.

5.2 Разработать подпрограмму на языке ассемблера, вычисляющую выражение $(a+b)*(c-d)$. Для передачи параметров a, b, c, d использовать стек. Результат нужно возвращать в вызывающий код.

5.3 Разработать подпрограмму на языке ассемблера, вычисляющую выражение ax^2+bx+c . Для передачи параметров a, b, c, x использовать регистры в соответствии с Windows ABI. В процессе вычислений для хранения промежуточных значений вычислений использовать локальные переменные в стеке. Результат нужно возвращать в вызывающий код.

5.4 Разработать подпрограмму на языке ассемблера, вычисляющую длину переданной в параметры строки (признаком конца строки считать символы `\0` и перенос строки) и возвращающую результат в вызывающий код.

Подпрограмму расположить в отдельном файле и подключать к вызывающему коду на языке ассемблера в процессе компиляции.

5.5 Разработать подпрограмму на языке ассемблера, сравнивающую два массива. Результат сравнения (массивы равны (1) или массивы не равны (0)) вернуть в вызывающий код. Подпрограмму расположить в отдельном файле и подключать к вызывающему коду на языке Си в процессе компиляции.

6 Порядок выполнения работы

6.1 Запустить IDE SASM

6.2 Выполнить все задания из п.5.

6.3 Ответить на контрольные вопросы.

7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

8 Контрольные вопросы

8.1 Как в программе на языке ассемблера объявить целое число, строку, массив?

8.2 Каков синтаксис вызова процедур в программах на языке ассемблера?

8.3 Какие способы передачи параметров между процедурами используются в программах на языке ассемблера?

8.4 Как скомпилировать программу, состоящую из нескольких модулей?