Министерство высшего образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ

УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

Кафедра безопасности информационных систем (БИС)

КОМБИНИРОВАННЫЕ ПРОГРАММЫ. СВЯЗЫВАНИЕ РАЗНОЯЗЫКОВЫХ МОДУЛЕЙ

Отчет по лабораторной работе №3

по дисциплине «Системное программирование»

Студентка гр.737-1

\_\_\_\_\_\_ Агеева В.С.

\_\_.\_\_.2021г

Принял

Руководитель

доцент кафедры БИС

\_\_\_\_\_\_\_ Романов А.С.

\_\_.\_\_.2021г

Томск 2021

1 Введение

Цель работы: познакомиться с основными способами передачи параметров подпрограмм, особенностями передачи управления между модулями, научиться писать комбинированные программы, в которых модули Ассемблера вызываются из модулей, написанных на высокоуровневых языках программирования.

Задание:

1. Изучить краткие теоретические сведения, материалы лекций по теме практического занятия и приведенные выше примеры программ.

2. Все действия, описанное далее выполнять в docker контейнере, образ - любой привычный linux.

3. Решение основной задачи реализовать на ассемблере в виде процедуры (рисунок 1.1). Далее получить объектный модуль. Вызывать из кода на C++ процедуру из объектного файла и скомбинировать их в один исполняемый файл.

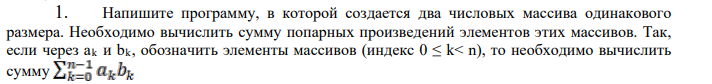


Рисунок 1.1 – Задание на лабораторную работу

2 Ход работы

Для выполнения лабораторной работы создадим 3 файла через nano:

- с кодом ассемблера;

- с кодом на высокоуровневом языке с использованием вставки ассемблера;

- докер (рисунок 2.1).

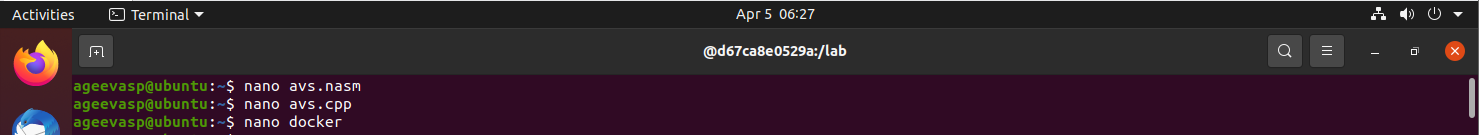


Рисунок 2.1 – Создание файлов

На рисунках 2.2 – 2.3 показан код ассемблера и код на высокоуровневом языке.

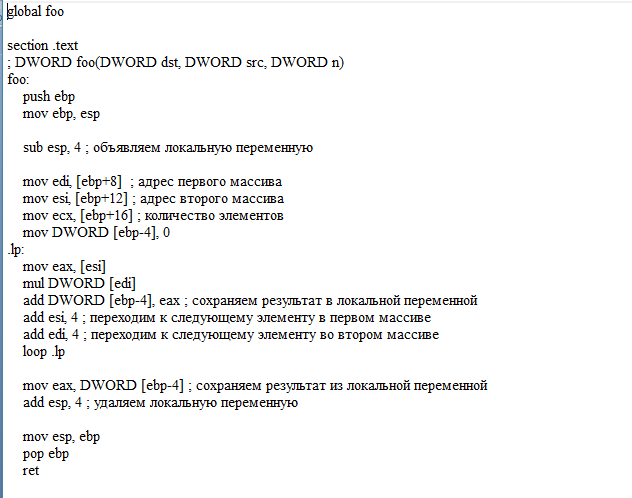


Рисунок 2.2 – Код на языке ассемблера

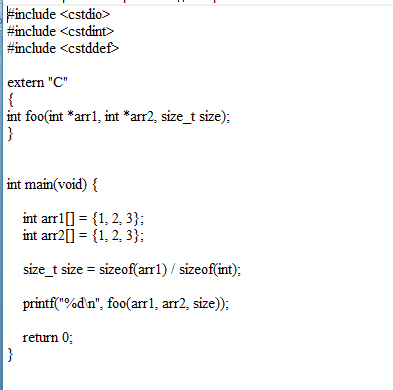


Рисунок 2.3 – Код на языке С

Соберем и запустим наш образ докер файла (рисунки 2.4 – 2.5).

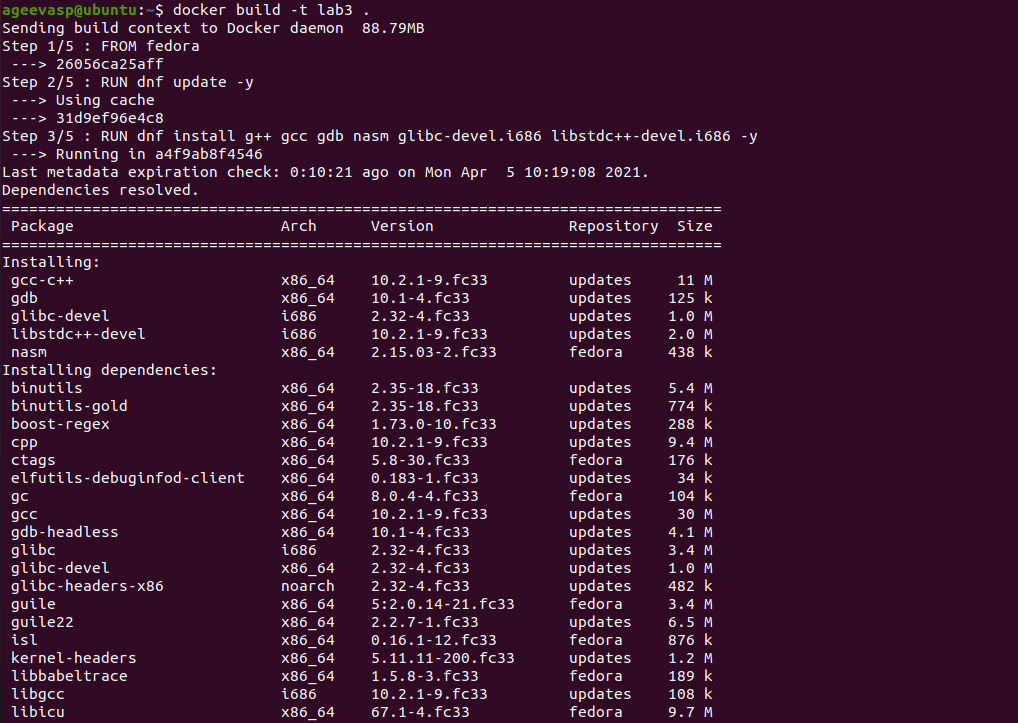


Рисунок 2.4 – Сборка докер файла

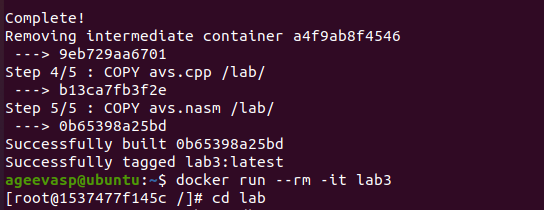


Рисунок 2.5 – Запуск докер файла

Для того, чтобы код на высокоуровнем языке программирования скомпилировался, необходимо ввести следующее: dnf install glibc-devel.i686 libstdc++-devel.i686.

На рисунке 2.6 показано какие нужно выполнять действия дальше.

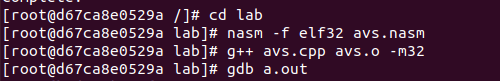


Рисунок 2.6 – Компиляция кодов и запуск gdb

3 Заключение

В ходе лабораторной работы были приобретены знания и навыки написания комбинированных программ, в которых модуль, написанный на ассемблере, вызывается программой, написанной на языке высокого уровня.

Для выполнения задания по варианту была написана и протестирована программа на ассемблере, затем адаптирована и интегрирована в виде ассемблерной вставки в программу, написанную на языке Си.

Исходные коды программ и Dockerfile приложены в архиве к отчету, а также загружены на GitHub ().