Лабораторная работа №8

Архитектура компьютера

Иванов Александр

Содержание

# 1 Цель работы

Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

# 2 Теоретическое введение

Стек — это структура данных, организованная по принципу LIFO («Last In — First Out»

или «последним пришёл — первым ушёл»). Стек является частью архитектуры процессора и реализован на аппаратном уровне. Для работы со стеком в процессоре есть специальные регистры (ss, bp, sp) и команды. Основной функцией стека является функция сохранения адресов возврата и передачи аргументов при вызове процедур. Кроме того, в нём выделяется память для локальных переменных и могут временно храниться значения регистров. На рис. 8.1 показана схема организации стека в процессоре. Стек имеет вершину, адрес последнего добавленного элемента, который хранится в ре- гистре esp (указатель стека). Противоположный конец стека называется дном. Значение, помещённое в стек последним, извлекается первым. При помещении значения в стек указа- тель стека уменьшается, а при извлечении — увеличивается. |

# 3 Выполнение лабораторной работы

Создание рабочего каталога и рабочего файла (рис. [1](#fig:001)).

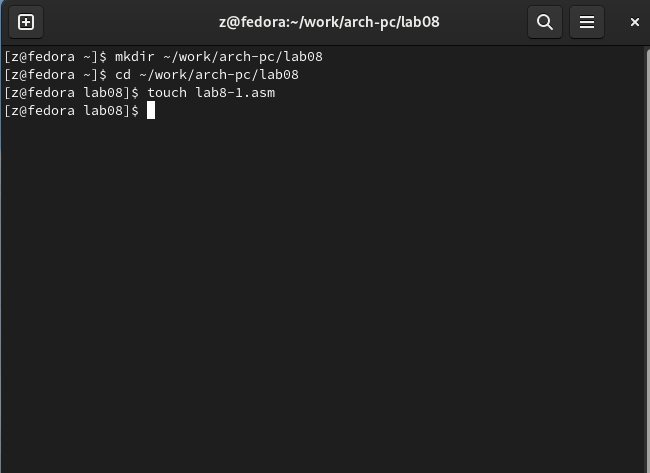


Figure 1: использование команды ‘mkdir’ и ‘touch’

Написание кода вывода чисел меньше переменной(рис. [2](#fig:002)).

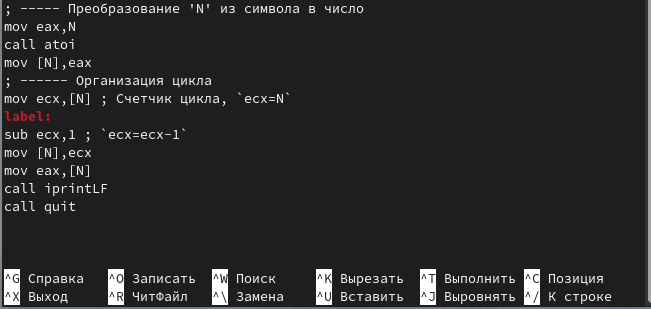


Figure 2: окно MidnightCommander

Создание и запуск программы(рис. [3](#fig:003)).

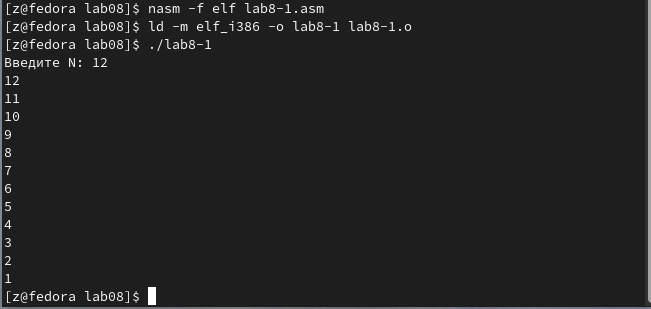


Figure 3: результат выполнения кода

Редактирование кода вывода числа меньше переменной на 1(рис. [4](#fig:004)).

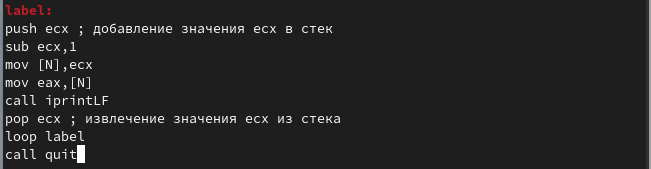


Figure 4: окно MidnightCommander

Создание и запуск программы(рис. [5](#fig:005)).

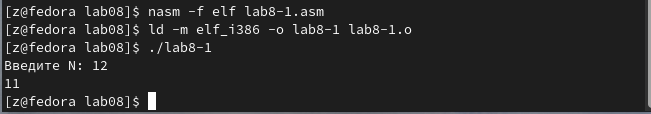


Figure 5: результат выполнения кода

Редактирование кода вывода чисел меньше переменной,(рис. [6](#fig:006)).

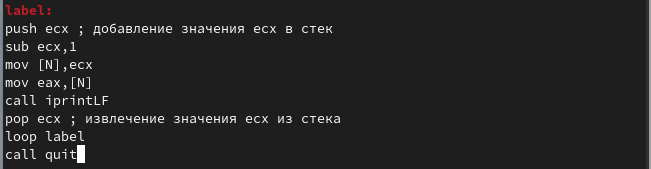


Figure 6: окно MidnightCommander

Создание и запуск программы(рис. [7](#fig:007)).

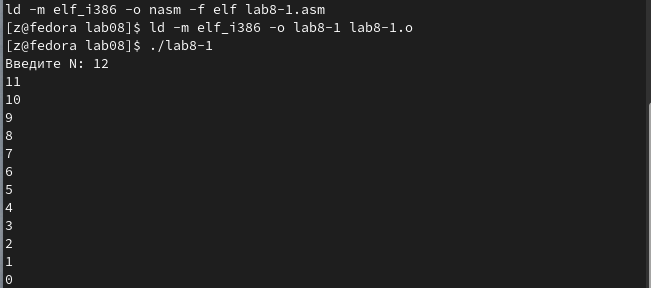


Figure 7: результат выполнения кода

Написание кода обработки аргументов командной строки(рис. [8](#fig:008)).

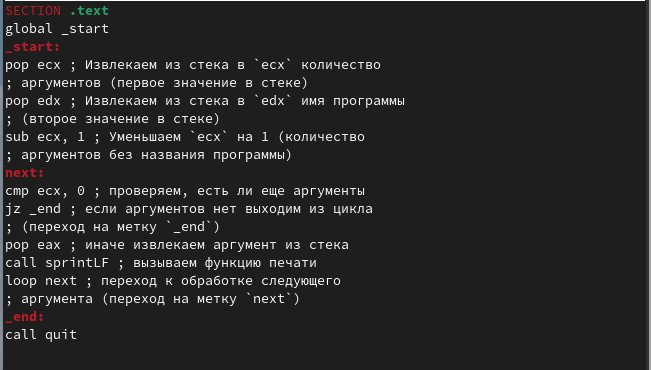


Figure 8: окно MidnightCommander

Создание и запуск программы(рис. [9](#fig:009)).

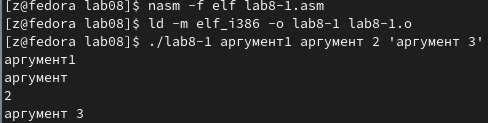


Figure 9: результат выполнения кода

Написание кода для вычесления суммы чисел меньше переменной(рис. [10](#fig:010)).

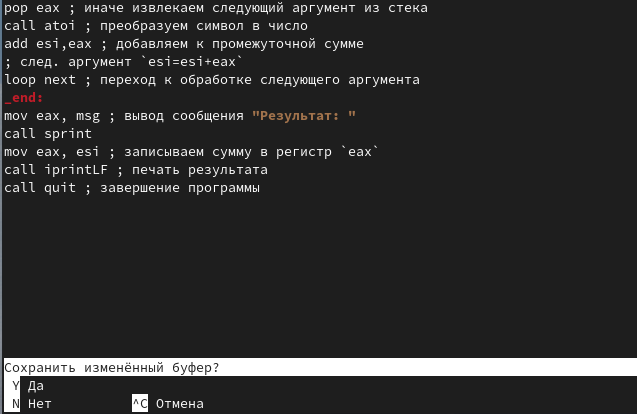


Figure 10: окно MidnightCommander

Создание и запуск программы(рис. [11](#fig:011)).

Figure 11: результат выполнения кода

Figure 11: результат выполнения кода

Написание кода для вычесления произведения чисел меньше переменной(рис. [12](#fig:012)).

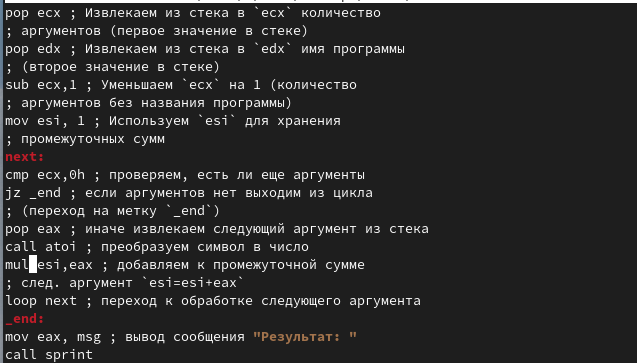


Figure 12: окно MidnightCommander

Создание и запуск программы(рис. [13](#fig:013)).

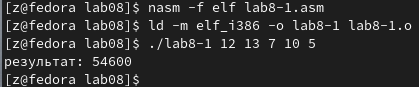


Figure 13: результат выполнения кода

# 4 Самостоятельная работа

Написание кода для вычесления суммы результатов функции(рис. [14](#fig:014)).

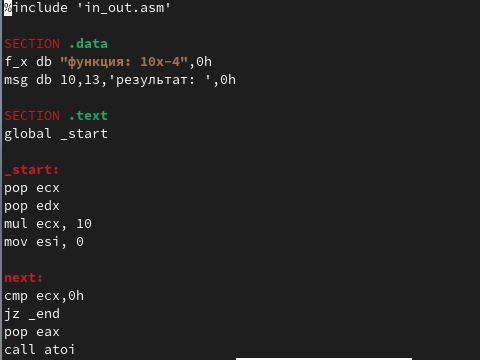


Figure 14: окно MidnightCommander

Создание и запуск программы(рис. [15](#fig:015)).



Figure 15: результат выполнения кода

# 5 Выводы

В ходе выполнения лабораторной работы я приобрел навыки написания программ с использованием циклов и обработкой аргументов командной строки

# Список литературы