

Мерочкин Илья БПИ218

Индивидуальное домашнее задание по архитектуре вычислительных систем №1

Вариант 6

6. Сформировать массив В, состоящий из элементов массива А, значения которых кратны введенному числу Х.

4 балла

В ассемблерную программу, откомпилированную без оптимизирующих и отладочных опций добавлены комментарии, поясняющие эквивалентное представление переменных в программе на С. Команда, которой это было сделано: `gcc -O0 -Wall -masm=intel -S asm.c -o asm.s`

Из ассемблерной программы убраны лишние макросы за счет использования соответствующих аргументов командной строки и за счет ручного редактирования исходного текста ассемблерной программы. Команда, которой это было сделано: `gcc -masm=intel -fno-asynchronous-unwind-tables -fno-jump-tables -fno-stack-protector -fno-exceptions asm.c -S -o asm.s`.

Тестовые прогоны для обеих программ:

Тест 1

Входные данные

1 2 3 4 5 6 7 8

2

Выходные данные программы на С

2 4 6 8

Выходные данные ассемблерной программы:

2 4 6 8

Тест 2

Входные данные

9 3 2 1 7 12 5

3

Выходные данные программы на С

9 3 12

Выходные данные ассемблерной программы:

9 3 12

Тест 3

Входные данные

1 2 3

4

Выходные данные программы на C

Выходные данные ассемблерной программы:

5 баллов

В реализованной программе используется функция `get_size` (считает и возвращает количество элементов, которое будет в массиве `b`) с передачей данных через параметры, с локальными переменными и возвращаемым значением. В комментариях ассемблерной программы описывается передача параметров и перенос возвращаемого значения.

6 баллов

Осуществлен рефакторинг программы на ассемблера, за счет использования регистров процессора, вместо обращения к памяти. Добавлены комментарии, поясняющие эквивалентное использование регистров вместо переменных на C.

Объектный файл программы с использованием регистров получилась немного меньше обычной. С бинарными файлами ситуация аналогичная.

7 баллов

Программа реализована в виде 2 единиц компиляции.

`asm.c` `get_size.c` - файлы с кодом на C. `asm.s` `get_size.s` - соответствующие ассемблерные программы.

Изначально файлы с кодом на C были отдельно откомпилированы в ассемблерные программы, а затем командой `gcc asm.s get_size.s -o ./asm` скомпилированы в исполняемый файл.

В программе присутствует файловый ввод/вывод. Имена файлов задаются с помощью аргументов командной строки. Вводятся два файла - для ввода и для вывода. Например, `./asm input.txt output.txt`. Программа проверяет на корректность число аргументов командной строки и корректное открытие файлов.

Также подготовлены файлы, обеспечивающие полное тестовое покрытие программы. В файлах `input_testN.txt` лежат входные данные, а в файлах `output_testN.txt` выходные для теста N. Запускается программа следующим образом: `./asm input_testN.txt output_testN.txt` (`./asm` - исполняемый файл).