

Отчёт по лабораторной работе №6

Дисциплина: Архитектура компьютера

Толстых Александра Андреевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Выполнение арифметических операций в NASM	9
3.2	Выполнение задания для самостоятельной работы	11
4	Выводы	12

Список иллюстраций

3.1	Создание каталога и файла	7
3.2	Написание программы	7
3.3	Копирование файла	7
3.4	Выполнение программы	7
3.5	Изменение программы	7
3.6	Выполнение программы	8
3.7	Создание файла и написание программы	8
3.8	Выполнение программы	8
3.9	Изменение программы	8
3.10	Выполнение программы	8
3.11	Изменение программы	8
3.12	Выполнение программы	9
3.13	Создание файла и написание программы	9
3.14	Выполнение программы	9
3.15	Изменение программы	9
3.16	Выполнение программы	9
3.17	Создание файла и написание программы	9
3.18	Выполнение программы	10
3.19	Создание файла и написание программы	11
3.20	Выполнение программы	11

Список таблиц

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

1. Символьные и численные данные в NASM.
2. Выполнение арифметических операций в NASM.
3. Выполнение задания для самостоятельной работы (16 вариантов).

3 Выполнение лабораторной работы

Создаю каталог для программ лабораторной №6. Перехожу в него и создаю файл lab6-1.asm (рис. 3.1).

Создание каталога и файла

Рис. 3.1: Создание каталога и файла

Ввожу в созданный файл текст программы из листинга (рис. 3.2).

Написание программы

Рис. 3.2: Написание программы

Копирую файл in_out.asm в каталог для программ лабораторной работы №6 (рис. 3.3).

Копирование файла

Рис. 3.3: Копирование файла

Создаю исполняемый файл и запускаю его (рис. 3.4).

Выполнение программы

Рис. 3.4: Выполнение программы

Изменяю текст программы, вместо символов записывая в eax, ebx числа (рис. 3.5).

Изменение программы

Рис. 3.5: Изменение программы

Создаю исполняемый файл и запускаю его (рис. 3.6).

Выполнение программы

Рис. 3.6: Выполнение программы

На экране ничего не отображается. Это связано с тем, что символ с кодом 10 - это символ перевода строки.

Создаю файл lab6-2.asm в каталоге для программ лабораторной №6. Ввожу в него текст программы из листинга 6.2 (рис. 3.7).

Создание файла и написание программы

Рис. 3.7: Создание файла и написание программы

Создаю исполняемый файл и запускаю его (рис. 3.8).

Выполнение программы

Рис. 3.8: Выполнение программы

Аналогично предыдущей программе заменяю символы на числа (рис. 3.9).

Изменение программы

Рис. 3.9: Изменение программы

Создаю исполняемый файл и запускаю его (рис. 3.10).

Выполнение программы

Рис. 3.10: Выполнение программы

Теперь программа складывает не коды, соответствующие символам, а сами числа. Поэтому выводит число 10 - сумму чисел 4 и 6.

Заменяю функцию `iprintln` на `iprint` (рис. 3.11).

Изменение программы

Рис. 3.11: Изменение программы

Создаю исполняемый файл и запускаю его (рис. 3.12).

Выполнение программы

Рис. 3.12: Выполнение программы

Вывод функции `iprintLF` от вывода функции `iprint` отличается тем, что в последнем случае после вывода не добавляется переход на новую строку.

3.1 Выполнение арифметических операций в NASM

С помощью утилиты `touch` создаю файл `lab6-3.asm`. Ввожу в него текст программы для вычисления значения указанного выражения (рис. 3.13).

Создание файла и написание программы

Рис. 3.13: Создание файла и написание программы

Создаю исполняемый файл и запускаю его (рис. 3.14).

Выполнение программы

Рис. 3.14: Выполнение программы

Изменяю текст программы для вычисления нового выражения (рис. 3.15).

Изменение программы

Рис. 3.15: Изменение программы

Создаю исполняемый файл и запускаю его (рис. 3.16).

Выполнение программы

Рис. 3.16: Выполнение программы

С помощью утилиты `touch` создаю файл `variant.asm`. Ввожу в него текст программы для вычисления варианта (рис. 3.17).

Создание файла и написание программы

Рис. 3.17: Создание файла и написание программы

Создаю исполняемый файл и запускаю его (рис. 3.18).

Выполнение программы

Рис. 3.18: Выполнение программы

Проверяю результат работы программы, вычисляя номер варианта аналитически. Также получаю число 16. Значит программа работает корректно.

Ответы на вопросы по программе: 1. За вывод сообщения “Ваш вариант” отвечают строки кода:

```
mov eax,rem  
call sprint
```

2. Инструкция `mov ecx, x` выполняется для того чтобы положить адрес вводимой строки `x` в регистр `ecx`. Инструкция `mov edx, 80` выполняется для записи длины вводимой строки в регистр `edx`. Инструкция `call sread` выполняется для вызова подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры.
3. Инструкция `call atoi` используется для вызова подпрограммы из внешнего файла, которая преобразует `ascii`-код символа в целое число и записывает результат в регистр `eax`.
4. За вычисление варианта отвечают строки:

```
xor edx, edx  
mov ebx, 20  
div ebx  
inc edx
```

5. При выполнении инструкции `div ebx` остаток от деления записывается в регистр `edx`.
6. Инструкция `inc edx` увеличивает значение регистра `edx` на 1.
7. За вывод на экран результата вычислений отвечаю следующие строки:

```
mov eax,edx  
call iprintLF
```

3.2 Выполнение задания для самостоятельной работы

Создаю файл lab6-4.asm и записываю в него программу для вычисления выражения 16 варианта. (рис. 3.19).

Создание файла и написание программы

Рис. 3.19: Создание файла и написание программы

Создаю исполняемый файл и проверяю его работу для указанных значений (рис. 3.20).

Выполнение программы

Рис. 3.20: Выполнение программы

4 Выводы

При выполнении данной лабораторной работы я освоила арифметические инструкции языка ассемблера NASM.