Отчёт по лабораторной работе №7

дисциплина: Архитектура компьютера

Сычев Егор Олегович

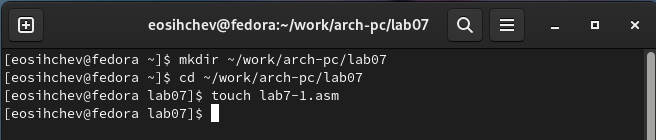
Содержание

# 1 Цель работы

Освоить арифметические инструкции языка ассемблера NASM.

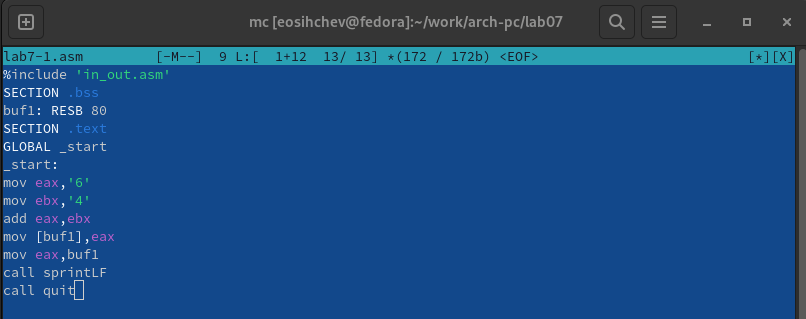
# 2 Выполнение лабораторной работы

1. Создаем каталог для программ лабораторной работы №7, переходим в него и создаем файл lab7-1.asm.



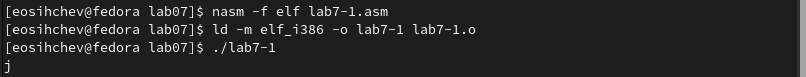
Команды: mkdir, cd, touch

1. Вводим в файл lab7-1.asm текст программы из листинга 7.1.



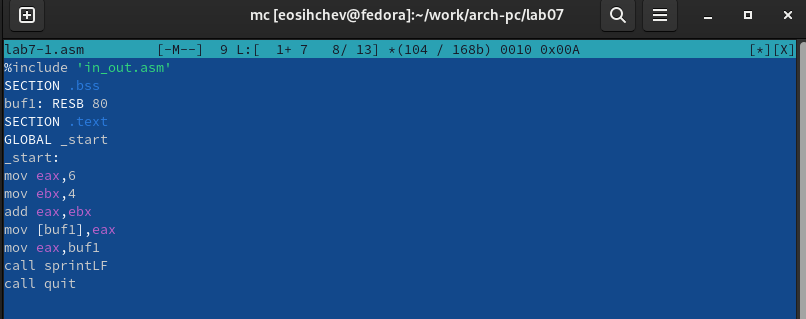
mcedit

1. Создаем испольняемый файл и запускаем его.

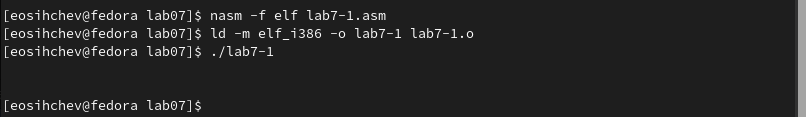


Транслирование, компоновка, запуск

1. Изменяем текст программы и вместо символов, запишем в регистры числа. Создаем исполняемый файл и запускаем его. В данном случае выводится символ с кодом 10 - перенос строки.



mcedit

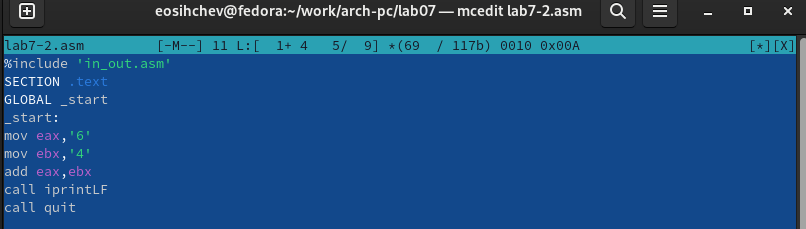


Транслирование, компоновка, запуск

1. Создаем файл lab7-2.asm, вводим в него текст программы из листинга 7.2.

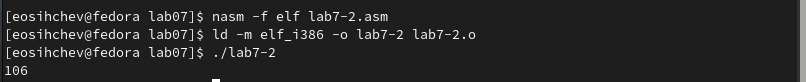
Команды: touch, mcedit

Команды: touch, mcedit



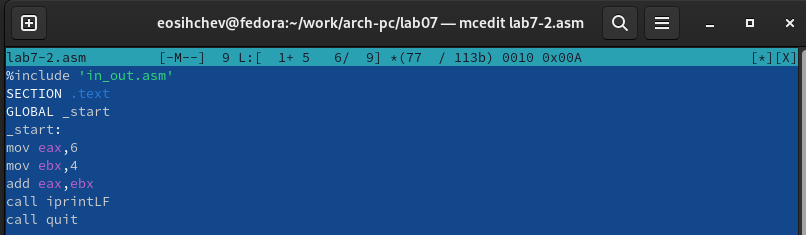
mcedit

1. Создаем испольняемый файл и запускаем его.

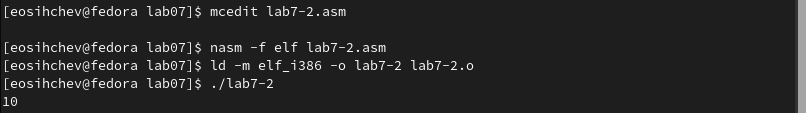


Транслирование, компоновка, запуск

1. Изменяем текст программы аналогично с lab7-1.asm. Создаем исполяемый файл и запускаем его.

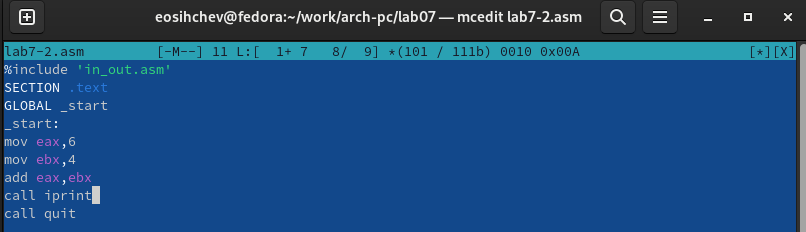


mcedit

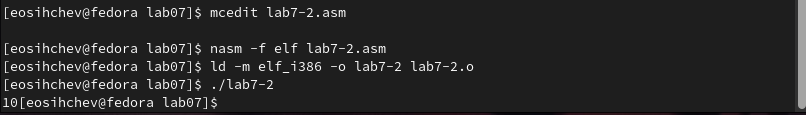


Транслирование, компоновка, запуск

1. Также заменяем функцию iptrintLF на iprint. Создаем исполяемый файл и запускаем его.



mcedit

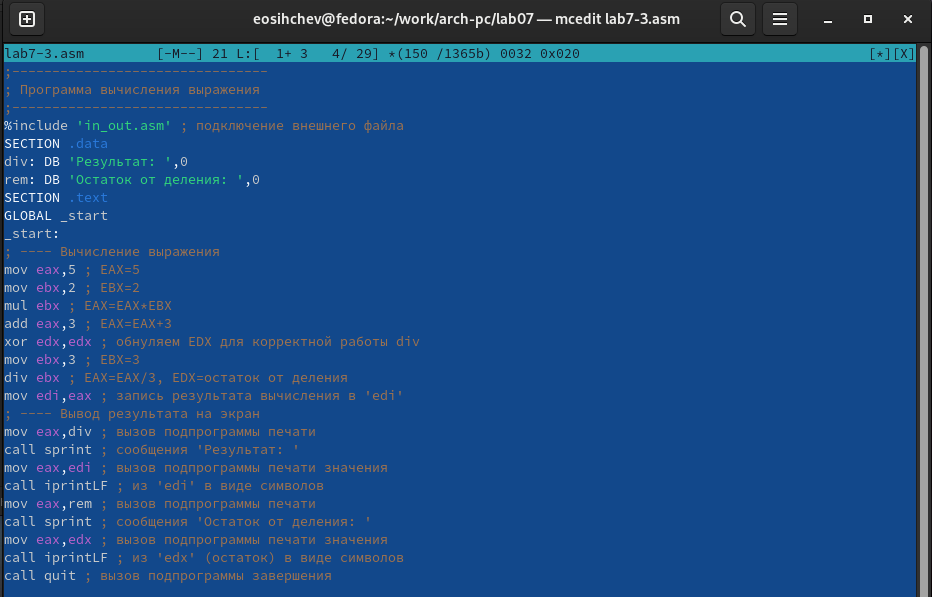


Транслирование, компоновка, запуск

1. Создаем файл lab7-3.asm, вводим в него текст программы из листинга 7.3.

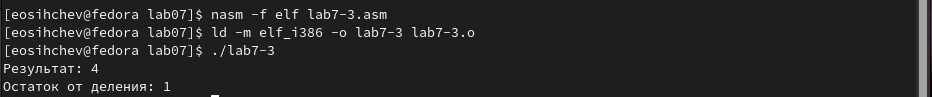
Команды: touch, mcedit

Команды: touch, mcedit



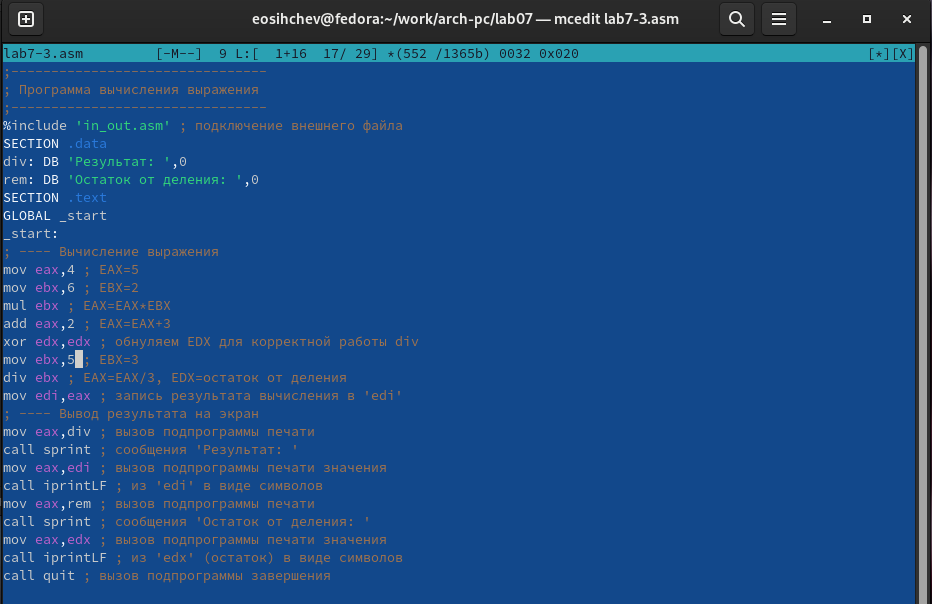
mcedit

1. Создаем испольняемый файл и запускаем его.

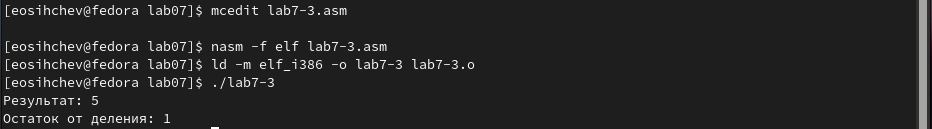


Транслирование, компоновка, запуск

1. Изменяем текст программы для вычисления выражения f(x)=(4\*6+2)/5. Создаем испольняемый файл и запускаем его.



mcedit

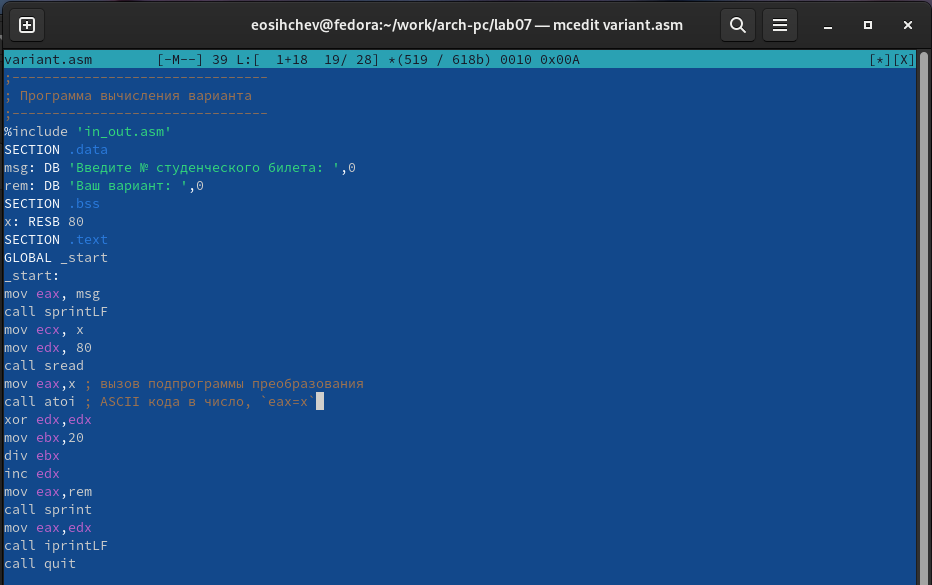


Транслирование, компоновка, запуск

1. Создаем файл variant.asm, вводим в него текст программы из листинга 7.4.

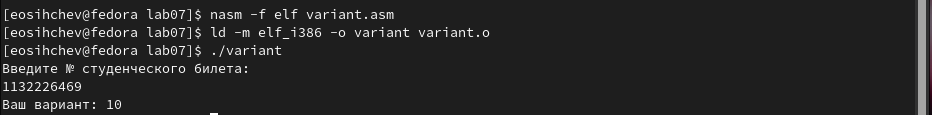
Команды: touch, mcedit

Команды: touch, mcedit



mcedit

1. Создаем испольняемый файл и запускаем его.



Транслирование, компоновка, запуск

# 3 Ответы на вопросы

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения ‘Ваш вариант’?

* mov eax,rem
* call sprint

1. Для чего используется следующие инструкции: mov eax, x | mov edx, 80 | call sread ?

* mov eax, x : записывает данные из x в регистр eax
* mov edx, 80 : указывает длину переменной x
* call sread : считывает введенную информацию

1. Для чего используется инструкция “call atoi”?

* Преобразует ASCII код в число

1. Какие строки листинга 7.4 отвечают за вычисления варианта?

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

1. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”?

* edx

1. Для чего используется инструкция “inc edx”?

* Прибавляет единицу

1. Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений?

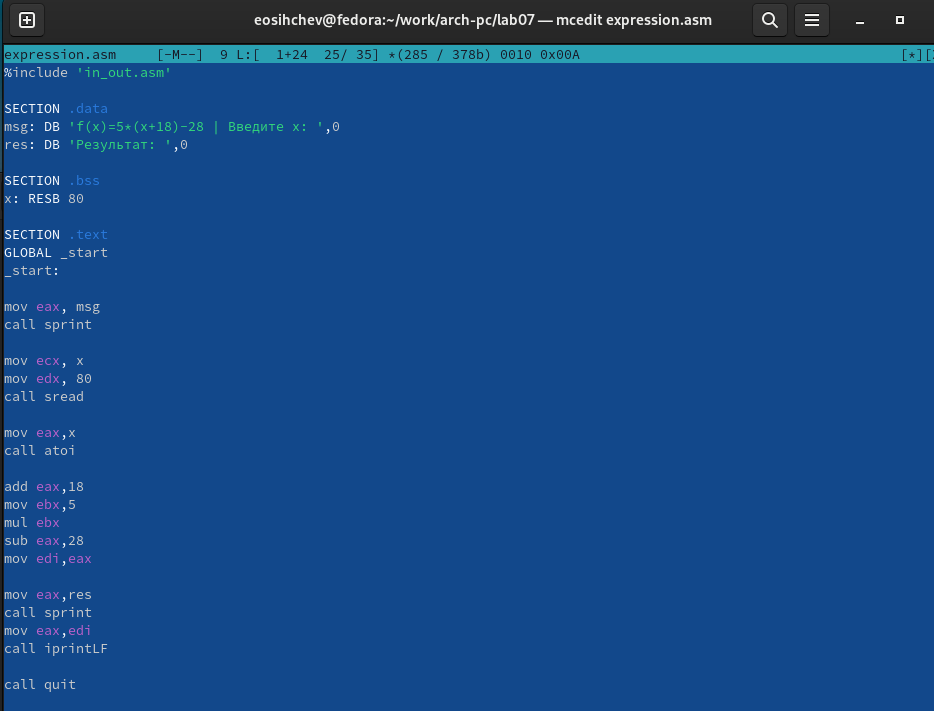
* mov eax,edx
* call iprintLF

# 4 Самостоятельная работа

1. Создаем файл expression.asm, вводим в него текст программы для вычисления выражения f(x)=5\*(x+18)-28.

Команды: touch, mcedit

Команды: touch, mcedit



mcedit

1. Создаем испольняемый файл и запускаем его. Для проверки возьмем 2 значения: x1=2 | x2=3.



Транслирование, компоновка, запуск

# 5 Вывод

Я освоил арифметические инструкции языка ассемблера NASM.