МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра вычислительной техники

**Лабораторная работа №4**

**по дисциплине «Технология программирования»**

|  |  |
| --- | --- |
| Студент | Ванин К.Е. |
| Группа | АВТ-819 |
| Преподаватель | Гаврилов К. В. |

Новосибирск

2020

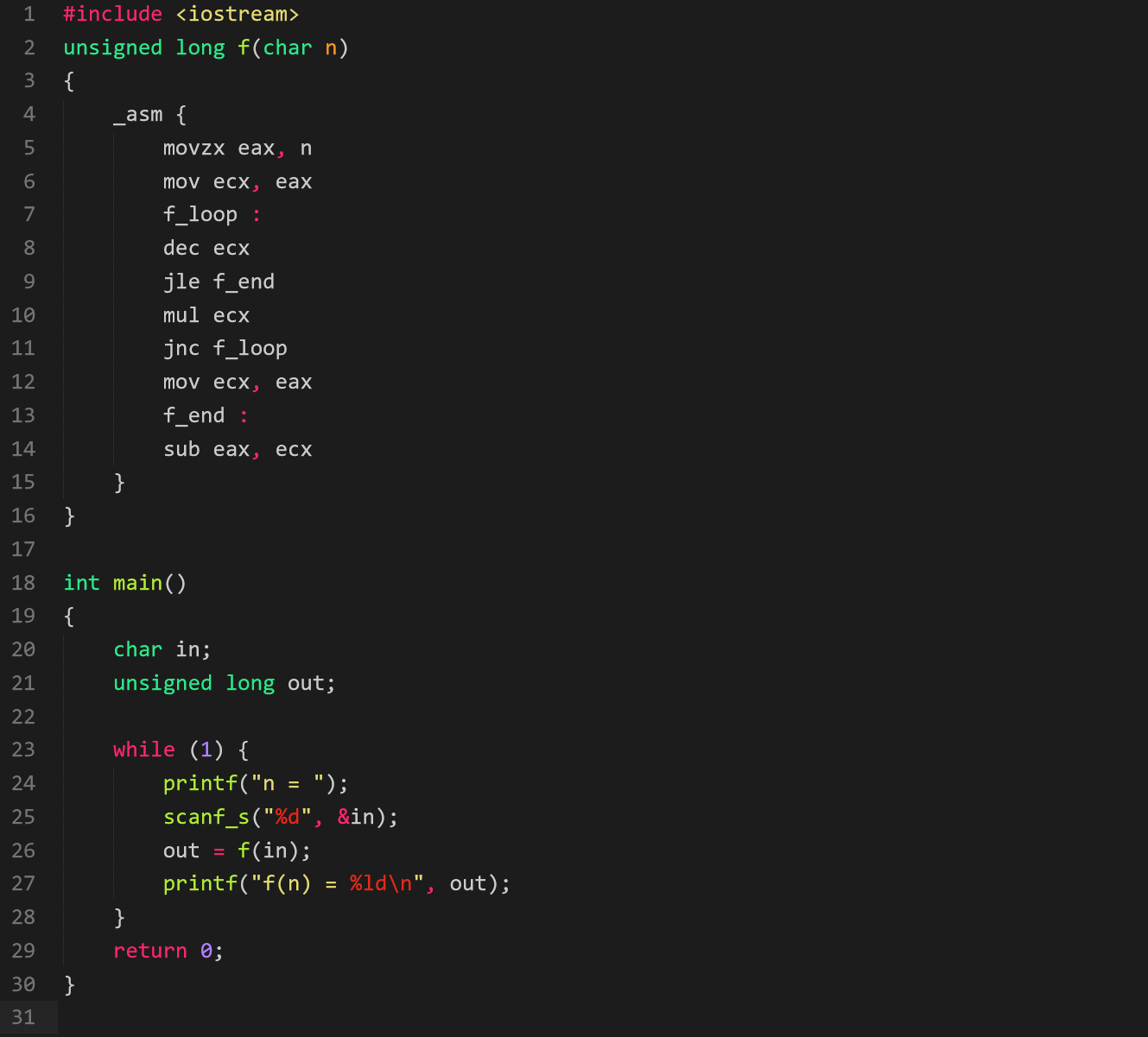
**Задание:**

1. Ввести с клавиатуры целое число от 0 до 255 и поместить его в переменную типа **char**

2. Вызвать для данной переменной функцию f, которая написана в виде ассемблерной вставки

3. Вывести результат на экран

**Листинг:**



**Результат выполнения программы:**

****

**Работа алгоритма:**

1. Команда **movzx** считывает значение **n** в регистр **eax**, при этом 0 расширяется (старшие **eax** 24 бита заполняем 0)
2. Полученное значение отправляется в **ecx**.
3. Значение регистра **ecx** уменьшается на 1. Если итоговое значение равно 0 => **ZF** = 1;

**SF** флаг содержит в себе получившееся значение. Если подается неотрицательное **n**, то в этой функции после декремента флаг переполнения будет всегда **OF** = 0.

1. Если **ZF** = 1 или **SF** не равно **OF** (значение **ecx** должно быть <= 0), то переходим по метке к пункту 8.
2. Команда **mul** умножает содержимое регистра **ecx** на содержимое **eax**. Младшие 32 бита результата отправляются в **eax**, а старшие 32 бита – в **edx**, однако если после этого значение регистра **edx** = 0 => флаг **CF** = 0, иначе **CF** = 1.
3. Если **CF** = 0 (число по прежнему вмещается в 32 бита), то переходим по метке к третьему пункту.

Если **CF** = 1 (результирующее число не помещается в 32 бита), то выполняем след. команду.

1. Заносится значение регистра **eax** в регистр **ecx**.
2. Вычитается значение регистра **ecx** из значения регистра **eax** и помещаем в регистр **eax**.

**Частный случай:**

1. При n >= 13, число не может быть представлено 32-разрядным целым во время одной из итераций цикла. Если так, то в пункте 6 цикл прерывается и исполняется 7 пункт, поскольку в 7 пункте устанавливается равные значения в регистрах **eax** и **ecx**. Результат => **eax** – **ecx** =0.
2. Если 1<= n <=12, то устанавливается значение **eax** = **n**. Заносим это значение в **ecx**, и уменьшаем его на 1.

Пока **ecx** != 0, умножаем его на содержимое **eax** (результат заносится в **eax**) и уменьшаем значение **ecx** на 1. Когда **ecx** = 0, переходим по метке к 8 пункту. Если **ecx** = 0, то результат равен числу, которое находится в **eax**.

1. Если n = 0, то в 3 пункте устанавливаются: значение **ecx** = -1, а флаги: **ZF и OF** = 0, **SF** =1 => переходим к 8 пункту. Результат равен **eax** -**ecx** = 0-(-1) = 1. Также, в 1 пункте значение eax устанавливается равным 0.