# Test Prime Numbers in Assembly

## Emmanuel Benoist, Christian Grothoff, Pascal Mainini

## Fall Term 2019/2020

This exercise must be done inside the clone of your own fork of the skeleton git repository.

## 1 Presentation of the files

The file `prime.asm` contains an assembler file that needs to be completed. This is the main file you need to change.

The file `makefile` contains the instructions for the compilation.

## 2 Goal of the exercise

The goal of this exercise is to work your first real Assembly language program. This is a program that reads a number given as parameter to the program and writes the number of prime numbers bellow this number.

For instance if the input is 23, the value returned is 8. This is for the numbers 2,3,5,7,11,13,17 and 1 (which is not a prime but we count it for this exercise).

If the input is 20, the value returned is also 8. If the input is 14, the output is 6 (for 1,2,3,5,7,11). If the input is 100, the output is 25

## 3 Expected Output

The program must have the following behaviour:

```
machine−:$ ./prime 23
8
machine−:$ ./prime 20
8
machine−:$ ./prime 14
6
machine−:$ ./prime 100
25
```

## 4 Upload the result to the Git repository

You must upload the file you wrote `primes.asm` to the repository.

```
machine−:$ git add primes.asm
....
machine−:$ git commit −m"solution for exercise 2"
....
machine−:$ git push origin master
```

## 5 Get feedback

You can inspect the results of the automated tests for your program inside gitlab CI (`https://gitlab.ti.bfh.ch`, go to `CI/CD > Jobs` in your forked project). There, you will also find the amount of points given.