LAB EXPERIMENT #2

Objective(s)

- Become familiar with using variables, branching, and arithmetic operations in 8086 emulator software.

Lab Work

Write an assembly program that is able to determine if a given number is (i) prime or not and (2) perfect square or not.

1. Define three variables: `number`, `isPrime`, and `isSquare`. The programmer assigns the value of variable `number`, and `isPrime` together with `isSquare` are set to 0 by default.
2. *" A natural number that possesses only two factors, itself and 1, is called a prime number."*
   You should create a loop that divides the `number` by 2, 3, …, (`number`-1) at each iteration. If the remainder after division is equal to 0 at any iteration, the number is not prime. So, you should break the loop and exit the program. If the loop completes, it means the number is prime indeed. Then, you should set isPrime as 1 and exit the program.
3. *"A square number or perfect square is an integer that is the square of an integer; in other words, it is the product of some integer with itself."*
   Similar to prime number investigation, check if `number` is square of 2, 3, …, sqrt(number). Set isSquare to 1 if `number` is square of a smaller number.

Hint-1: You can see the usage of DIV instruction on the emulator's help page.
Hint-2: You can check the remainder by CMP instruction and jump to a point at the code according to JE instruction's result (See JE instruction usage at the emulator's help page.)

Evaluation: You must complete your experiment before the lab hour.