Program 1: check if a number is prime

```
GO
var n@Int;
read(n);
var isPrime@Bool = True;
if n < 2{
        isPrime = False;
}
elif n == 2{
        isPrime = True;
}
elif n % 2 == 0{
        isPrime = False;
}

var d@Int = 3;

loop d <= n/2 {
        if n % d == 0:
                isPrime = False;
        d = d + 3;
}
isPrime = True;
STOP
```

Program2: read a number and print the last digit

```
GO
var n@Int;
read(n);
let lastDigit@Int = n % 10;
print(lastDigit);
STOP
```

Program 3: compute the number of digits of a given number

```
GO
let count@Int = 0;
let n@Int = 432567;

loop n != 0 {
        n = n / 10;
        count = count + 1;
```

```
}

print(n);
STOP
```

Program1 error:

```
GO
var 2n@Int; // lexical error
read(n);
var isPrime@Bool = True;
if n < 2{
        isPrime = False;
}
elif n == 2{
        isPrime = Tru; // lexical error
}
elif n % 2 == 0{
        isPrime = False;
}

var d@Int = 3;

loop d <= n/2 {
        if n % d == 0:
                isPrime = False;
        d = d + 3;
}
isPrime = True;
STOP
```