

Lab 1a

**Statement:** Considering a small programming language (that we shall call mini-language), write 3 small programs in this language.

**Deliverables:** p1.\*, p2.\*, and p3.\* and p1err.\* - small programs written in your programming language (p1, p2, p3 should be lexically correct; p1err should contain 2 types of lexical errors).

**For example:**

**p1 and p2:** compute the max/min of 3 numbers; verify if a number is prime, compute gcd of 2 numbers, compute the solutions for a 2nd order equation, aso

**p3:** compute the sum of n numbers, compute the max/min of n numbers

p1.lng

//check if a number is prime

```
mainprogram(){
    defINT a;
    defBOOL prime;
    defSTRING message;
    input(a);
    prime = true;
    if(a < 2)
        prime = false;
    if(a > 2 && a % 2 == 0)
        prime = false;
    for(defINT i = 3; i * i <= a; i = i + 2)
        if(a % i == 0)
            prime = false;
    if(prime == true)
        message = "the number is prime";
```

```
        else
            message = "the number is not prime";
        output(message);
    }
```

p2.lng

//compute the maximum of 3 numbers

```
mainprogram(){
    defINT a, b, c, maximum;
    defSTRING message;
    input(a);
    input(b);
    input(c);
    maximum = a;
    message = "first number is the biggest number";
    if(b > maximum){
        maximum = b;
        message = "second number is the biggest number";
    }
    if(c > maximum){
        maximum = c;
        message = "third number is the biggest number";
    }
    output(message);
    output(maximum);
}
```

p3.lng

//compute the sum of an array

```

mainprogram(){
    defFLOAT[] array = [1.2, 2, 3.4, 4, 7.5, 6, 7, 814.2, 9, 10];
    defFLOAT sum;
    sum = 0;
    for(defINT i = 0; i < array.length(); i++)
        sum = sum + array[i];
    output(sum);
}

```

p4.lng

//lexical error

//compute the sum of two numbers

```

mainprogram(){
    defINT 1a, a2;
    1a = 3;
    a2 = 2"3;
    defINT sum;
    sum = 0;
    sum = 1a + a2;
    output(sum);
}

```