- **p1 - compute gcd of 2 numbers**

@

```
DECLARE a integer;
DECLARE b integer;

READ(a);
READ(b);

WHILE (a != b):
        IF (a > b):
                a = a - b;
        ELSE:
                b = b - a;
WRITE(a);
```

@

- **p2 - compute lcm of 2 numbers**

@

```
DECLARE a integer;
DECLARE b integer;

READ(a);
READ(b);

DECLARE copy_a integer;
DECLARE copy_b integer;

copy_a = a;
copy_b = b;

DECLARE gcd integer;
DECLARE lcm integer;

WHILE (a != b):
        IF (a > b):
                a = a - b;
        ELSE:
                b = b - a;
gcd = a;

lcm = copy_a*copy_b / gcd;

WRITE(lcm);
```

@

- **p3: compute the sum of n numbers**

```
@
    DECLARE n integer;
    READ(n);

    DECLARE arr[n] array[integer];

    FOR i in (0:n):
        READ(arr[i]);

    DECLARE sum integer;
    sum = 0;

    FOR i in (0:n):
        sum += arr[i];

    WRITE(sum);
@
```

p1err should contain 2 types of lexical errors
p1err: Check if two chars are identical

```
@

    DECLARE 1char char;
    DECLARE char2 char;

    1char = ‘c’;
    char2 = ‘c;

    IF (1char == char2):
        WRITE(‘They are equal’);
    ELSE:
        WRITE(‘They are not equal’);
@
```