

p1: compute the minimum of 3 numbers

GO

```
have a be integer;
have b be integer;
have c be integer;
have min be integer;
read(a);
read(b);
read(c);
min = a;
assuming (min > b) then
    min = b;
stopAssuming;
assuming (min > c) then
    min = c;
stopAssuming;
write(min);
```

STOP

p2: verify if a number is prime

GO

```
have a be integer;
have prime be boolean;
have i be integer;
read(a);
assuming (a == 2) then
    prime = true;
otherwise
    assuming (a < 1 || a % 2 == 0) then
        prime = false;
    otherwise
        prime = true;
        for (i = 3; i * i <= a && prime; i = i + 1) do
            assuming (a % i == 0) then
                prime = false;
            stopAssuming;
        stopFor;
    stopAssuming;
stopAssuming;
write(prime);
```

STOP

p3: compute the sum of n numbers

GO

```

    have sum be integer;
    have n be integer;
    have i be integer;
    sum = 0;
    read(n);
    have numbers be integer[n];
    for (i = 0; i < n; i = i + 1) do
        read(numbers[i]);
        sum = sum + numbers[i];
    stopFor;
    write(sum);
STOP

```

p1err: compute the minimum of 3 numbers (with 2 types of lexical errors)

```

GO
    have 1a be integer;
    have b be integer;
    have c be integer;
    have min be integer;
    read(1a);
    read(b);
    read(c);
    min = 1a;
    assuming (min # b) then
        min = b;
    stopAssuming;
    assuming (min # c) then
        min = c;
    stopAssuming;
    write(min);
STOP

```