In [3]:

```
average <- function(){
    a <- as.double(readline())
    b <- as.double(readline())
    c <- as.double(readline())
    y = (a+b+c)/3
    print(paste(y," is average of ",a," ",b," ",c))
}
average()
```

```
[1] "2  is average of  1   2   3"
```

In [5]:

```
factorial <- function(n){
     fact = 1
    while(n>0){
        fact = fact*n
        n = n-1
    }
    return(fact)
}
factorial(5)
```

120

In [20]:

```
hl <- function(x,y){
    if(x > y) {
        smaller = y
    } else {
        smaller = x
    }
    for(i in 1:smaller) {
        if((x %% i == 0) && (y %% i == 0)) {
            hcf = i
        }
    }
```

```
    lcm = x*y/hcf
    return(list(lcm,hcf))
}
hl(2,6)
```

1. 6
2. 2

In [23]:

```
seq <- function(n){
    if(n==0){
        return(0);
    }
    return(seq(n-1)+n)
}
seq(5)
```

15

In [37]:

```
rev = 0
reverseofnum <- function(n){
#     if(n==0){
#         return(0);
#     }

    while (n > 0) {
      r = n %% 10
      rev = rev * 10 + r
      n = n %/% 10
    }
    #reverseofnum(n)
    return(rev)
}
reverseofnum(223)
```

322

In [54]:
```
Si <- function(x){
    si = (x[1] * x[2] * x[3])/100
    return(si)
}
x <- readline()
x <- strsplit(x," ")
x <- as.integer(x[[1]])
Si(x)
```

0.24

In [39]:
```
dtob <- function(n) {
    if(n > 1) {
    dtob(as.integer(n/2))
    }
    cat(n %% 2)
}
dtob(52)
```

110100

In [40]:
```
fact = 1
factorial <- function(n){
    if(n==0) return(1)
    return(factorial(n-1)*n)
}
factorial(5)
```

120

In [43]:
```
sumofs <- function(n){
    if(n==0) return(0)
    return(sumofs(n-1)+n**2)
}
sumofs(10)
```

385

In [55]:
```
xx <- function(x){
    sum = sum(x[1:5])
    print(sum)
    avg = sum/5
    print(avg)
    Si = round(sd(x[1:5]),2)
    print(Si)
}
x <- readline()
x <- strsplit(x," ")
x <- as.integer(x[[1]])
xx(x)
```

```
[1] 15
[1] 3
[1] 1.58
```

Questions

In [56]:
```
cumsum(c(1:5))
```

1. 1
2. 3
3. 6
4. 10
5. 15

In [58]:
```
rev(c(1:10))
```

1. 10
2. 9
3. 8
4. 7
5. 6
6. 5
7. 4
8. 3
9. 2
10. 1

In [59]:
```
rev(cumsum(c(1:10)))
```

1. 55
2. 45
3. 36
4. 28
5. 21
6. 15
7. 10
8. 6
9. 3
10. 1

In [62]:
```
sample(1:100,10)
```

1. 55
2. 33
3. 63
4. 66
5. 7
6. 88
7. 95
8. 58
9. 53
10. 86

In [63]:
```
log10(sqrt(100))
```

1

In [ ]:
```

```