

```
int x;
int y;
```

برنامه ای بنویسید که عدد را دریافت کند و آن را محاسبه کند
برنامه ای که آن بار تکرار شود

```
cin >> x;
cin >> y;
```

حل استاد

```
int i = 0;
int x, y;
int z = 1;
cin >> x >> y;
```

5

```
while (y != 0)
{
```

```
while (i < y)
{
```

تعداد آن بار اجرا می شود

```
cout << "1";
}
```

```
z = z * x;
```

```
i = i + 1;
```

```
cout << z;
```

10

```
while (x <= y)
{
    x = x + 1;
    y = y + 1;
    cout << "xy";
}
```

15

نکته: همه مباحث که مقادیری به آن از کاربرد داشته اند این یعنی نه هم بخاطر همین چند متغیر جدید تعریف می کنیم.

20

```
#include <math>
```

تولید ریاضی در برنامه نویسی

$y = \log(x)$; $\xrightarrow{\text{معادله جبر}} \ln \rightarrow \log$ طبیعی

25

```
cout << "y";
```

$y = \log_{10}(x)$; $\rightarrow \log$

جبر و \sqrt{x} $y = \sqrt{x}$

$y = \text{Pow}(,)$; توان



```

n      log
-----
1      ( )
2      ( ) # include <iostream.h>
2      ( ) # include <math.h>
4      ( )
:      ( )
:      ( )
:      ( )
100    ( )
    
```

using namespace std;

int main ()

{

int number = 0;

cout << "number > = 1 ++ number = < 100 }

کارین: برنامه این رو بنویسید.

cout << "the log of " << number << " is " << log (number);

get chyr ();

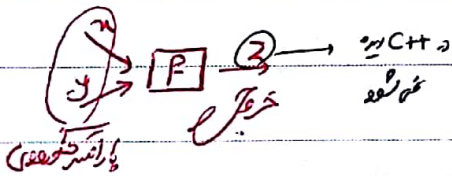
return 0;

1) برنامه بزرگ را با تغییرات به قطعات کوچکتر تقسیم کنیم

2) استفاده مجدد reuse

Function: خاصیت (ویژگی)

عنوان تابع (نیت پارامترها) Type name: تعریف تابع (انتقال تابع)



```

int sum (int x , int y)
{
    int z ;
    z = x + y ;
    return z ;
}
    
```

int z ;

z = x + y ;

return z ;

Definition

body

main

```

{
    int n;           // فراخوانی call
    n = sum(2, 3);
    cout << sum(5, 7);
    n = 2 * sum(2, 5) + 7;
}
    
```

توجه: استفاده از تابع

5

مثال: $n \rightarrow \boxed{n!}$

10 اولین مرحله: عنوان تابع

int fact (int n)

```

{
    int i = 1;
    int x = 1;
    while (i <= n)
    {
        x = i * x;
        i = i + 1;
    }
    return x;
}
    
```

Factorial

$$n! = n(n-1) \dots x1$$

پس می بینیم که $n!$ روی این به یک $n \times (n-1) \dots$ به معنای

15

0) cin در تابع تابع

main

```

{
    int n;
    cin >> n;
    cout << fact(n);
}
    
```

کامپایلر متوجه می شود

20

25 برای نام آن بنویسید که در m و n را از کاربر بگیرد (n, m) را با یکدیگر

$$\binom{m}{n} = \frac{m!}{n!(m-n)!}$$

→ fact reuse

```
int Fact (int n)
```

```
{
    int n = 1;
    int i = 1;
    while (i <= n)
    {
        n = i * n;
        i = i + 1;
    }
    return n;
}
```

Definition of Fact

5

```
main ( )
```

```
{
    int m;
    int n;
    cin >> m;
    cin >> n;
    int t;
    t = Fact(m) / (Fact(n) * Fact(m-n));
    cout << t;
}
```

10

15

20

برنامه الی بنویسید که بره های مساوی $ax^2 + bx + c = 0$ را پیدا کند.

```
#include <iostream>
using namespace std;
```

25

```
int main ( )
{
    int a, b, c, Δ;
}
```

