

Basic input/output - I/O - دستورات ورودی خروجی در برنامه نویسی C++

هدف از ساخت کامپیوتر و ایجاد برنامه نویسی دادن اطلاعات به ماشین و دریافت جواب بوده که به این روال، جریان ورودی خروجی گفته میشود. برای استفاده از این جریانات ما حداقل به ۲ فایل کتابخانه ای در C++ نیاز داریم و دستوراتی برای استفاده از این کتابخانه ها که با ارائه یک مثال، استفاده از این دستورات را نشان می دهیم.

```

1 #include <iostream.h>
2 #include <conio.h>
3 int main()
4 {
5     int a;           //a=?
6     cin >> a;       //a=value of user input
7     cout << "a:";  //Prints a: on screen
8     cout << a;     //Prints value of a
9     getch();
10    return 0;
11 }
```

a:120

برای استفاده از کلمات کلیدی cin و cout لازمه که از کتابخانه iostream.h در برنامه خود استفاده کنیم.

دستور cout باعث چاپ اطلاعات در مانیتور می شود به این صورت که اگر بعد از این دستور، عبارت مورد نظر رو تو " " قرار بدیم عینا همون عبارت تو مانیتور نشان داده می شود و معمولا جایی که کاربر قصد نشان دادن عین یک عبارت را در خروجی دارد مورد استفاده قرار می گیرد و اگه بخواهیم مقدار یک متغیر یا ثابتی را نمایش بدهیم باید نام را نوشته و از " " استفاده نکنیم.

دستور cin باعث می شود که از کاربر یا برنامه دیگه ای یک مقدار یا متنی را بگیریم. در واقع کامپایلر با رسیدن به این دستور منتظر ورود اطلاعات از طرف کاربر شده تا زمانی که کاربر Enter را فشار بده و مقدار ورودی کاربر را در متغیری که در جلوش اعلان شده بریزه و همونطور که قبلا اشاره شد موجب مقدار دهی به متغیر مربوطه میشود.

☆ در برنامه بالا تابعی بنام getch() هست که برای استفاده از اون باید از کتابخانه conio.h استفاده کنیم که باعث میشود بعد از اجرای برنامه و رسیدن کامپایلر به این خط، برنامه متوقف شده و منتظر این باشه که کاربر یک کلید از کیبورد را فشار دهد. با این دستور می توانیم خروجی برنامه را ببینیم چون اگر این تابع نباشد در کسری از ثانیه برنامه اجرا و بلافاصله بسته میشود و دیگر ما قادر به دیدن خروجی نخواهیم بود. (امتحان کنید!)

در مثال بالا فرض بر اینه که کاربر مقدار ۱۲۰ را وارد کرده است.

گاهی اوقات لازمه که ما چند مقدار را با استفاده از دستور cin به برنامه بدهیم، در اینصورت به ازاء هر مقدار از یک << استفاده می کنیم و همینطور برای دستور cout از یک >> .

```

int a, b;           //a=?, b=?
cin >> a >> b;
cout << a << b;
```

همونطور که در برنامه های قبلی دیدیم با نوشتن چند دستور چاپ، تمامی اطلاعات در یک خط نوشته شد. در خروجی برای اینکه از خط کنونی به خط بعدی برویم به یکی از صورتهای زیر عمل می کنیم:

```
cout << a << endl;
```

عبارت endl به کامپایلر پایان خط جاری را نشان می دهد و در صورت وجود خروجی دیگه ای در برنامه، نمایش خروجی را از خط بعدی ادامه میدهد.

```
cout << a << "\n";
```

عبارت \n نیز مثل endl خروجی را به خط بعدی هدایت می کند.

```

1  #include <iostream.h>
2  #include <conio.h>
3  int main()
4  {
5      int a, b;                //a=?,b=?
6      cout << "Please inter value of a:";
7      cin >> a;                //a=value of user input
8      cout << "Please inter value of b:";
9      cin >> b;                //b=value of user input
10     cout << "a:" << a << "\n"; //Prints a:(value of a) on screen and go to next line
11     cout << "b:" << b;        //Prints b:(value of a) on screen
12     return 0;
13 }
```

```

Please inter value of a:
Please inter value of b:
a:120
b:87
```

باید بگم فرض بر این است که مقادیر ۱۲۰ و ۸۷ توسط کاربر وارد شده است . نکته ای که اینجا قابل توجه است اینه که دستور cin خودش باعث میشود که کامپایلر با گرفتن مقدار بطور خودکار به خط بعدی برود و دیگه اینجا نیازی به \n یا endl نیست.