

فصل پنجم – ورودی و خروجی

- تابع خروجی (printf : در C)
- تابع ورودی (scanf: در C)
- خواندن اطلاعات از صفحه کلید با cin (در C++)
- چاپ اطلاعات با cout (در C++)
- خواندن کاراکترها از صفحه کلید با متدهای cin.get()
- نوع داده ای رشته ای
- ورودی و خروجی رشته ها

تابع خروجی (در C) : printf

برای این دستور از فایل سربار #include <stdio.h> استفاده می کنیم

printf ("...", ...);

نکته ۱) عبارت رشته ای می تواند کاراکترهای فرمت ، کاراکترهای کنترلی و یا یک متن و یا ترکیبی از آنها می باشد.

نکته ۲) کاراکترهای فرمت با % شروع می شوند و نوع متغیر را که در خروجی نمایش داده می شود را مشخص می کنند.

نکته ۳) کاراکترهای کنترلی با \ (بک اسلش) شروع می شوند و باعث انتقال مکان نمایش یک سطر و یا ستون خاص می شوند.

مثال printf(" this is the map ");

◀ معولی

مثال int a=10;

printf("%d", a);

◀ دارای کاراکتر فرمت

جدول کاراکترهای فرمت

کاراکتر فرمت	عملکرد
%d %i	برای چاپ اعداد صحیح مثبت و منفی
%ld	برای چاپ عدد صحیح طولانی (long)
%c	برای چاپ کاراکتر
%f %g %G	برای چاپ اعداد ممیز شناور (اعشاری)
%o	برای چاپ اعداد در مبنای اکتاو (8Octal)
%x %X	برای چاپ اعداد در مبنای هگزا (16Hex)
%s	برای چاپ رشته
%e	برای نمایش اعداد در مبنای علمی
%u	برای نمایش اعداد بدون علامت
%p	برای چاپ اشاره گر

جدول کاراکترهای کنترلی

کاراکتر کنترلی	عملکرد
\n	برای انتقال مکان نما در خروجی به خط بعدی
\t	برای انتقال مکان نما به اندازه یک Tab
\0	نتهای رشته را مشخص می کند.
\'	برای چاپ کوتیشن
\`	برای چاپ تک کوتیشن
\`	برای چاپ بک اسلش
\w	انتقال مکان نما به ۸ خط بعد
\b	انتقال مکان نما به یک کاراکتر قبل
\f	انتقال مکان نما به صفحه بعد

مثال (

```
int a=10;
```

```
float f=2.25;
```

```
char c='A' ;
```

```
printf(" %d  %f %c ",a ,f ,c);  ➔  10 2.25 A
```

```
printf(" a=%d\n  f=%f\n  C=%c " , a,f,c );
```

➔ a=10

b=2.25

c=A

```
printf("\\" this is the Avrage of Student \" = %d",a);
```

➔ " this is the Avrage of Student " = 10

```
printf("a=%d \n c=%c",a,c);
```

➔ a=10

c=A

سوال: حاصل دستور مقابل کدام است؟

چاپ اطلاعات با cout (در C++)

در زبان C++ برای چاپ یک متغیر یا یک عبارت از cout استفاده می‌کنیم. برای استفاده از این دستور از فایل سریاره #include <iostream.h> استفاده می‌کنیم.

نحوه استفاده از cout :

cout << عبارت آخر << ... << عبارت سوم << عبارت درم << عبارت اول <<

هر یک از عبارت‌ها می‌تواند یا اسم یک متغیر و با ترکیبی از کاراکترهای کنترلی و متن باشند. استفاده از کاراکترهای فرمت بی معنی می‌باشد.

```
cout<<"\n in the name of God\"";
```

→ "in the name of God"

در زبان C++ به جای "\n" می‌توان endl را نیز بکار برد.

```
cout<<"a=" << a << endl <<"c=" << c;
```

→ a=10

c=A

تابع ورودی (C در `scanf`)

تابع `scanf`

برای خواندن متغیرها از ورودی ، از این تابع استفاده می شود. در هنگام استفاده از `scanf` بجای نام متغیرها باید آدرس متغیرها بصورت زیر تعیین می شود.

`scanf ("...، آدرس متغیر دوم، آدرس متغیر اول، "کاراکترهای فرمت");`

آدرس متغیر `a` ← `&a` ← اسم متغیر + عملکر `&`

فرمت	عملکرد
<code>%c</code>	برای خواندن یک کاراکتر
<code>%d</code>	برای خواندن یک عدد صحیح
<code>%f</code>	برای خواندن یک عدد اعشار
<code>%s</code>	برای خواندن خواندن یک رشته
<code>%ld</code>	خواندن یک متغیر <code>long</code>

`scanf(آدرس متغیرها، "کاراکترهای فرمت");`

مثال ۱) **int a;**

```
scanf("%d", &a);
```

مثال ۲) **char c;**

```
scanf("%c", &c);
```

مثال ۳) **int a,b;**

```
float f;
char c;
scanf("%d%d%f%c",&a,&b,&f,&c);
```

در C++ می توان از کلاس **cin** به جای **scanf** استفاده کرد. که احتیاجی به آدرس دهی ندارد. **cin** در فایل سرباره **#include<iostream.h>** قرار دارد.

```
int a,b;    float f;    char c;
cin >> a >> b >> f >> c;
```



```
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string name;
    int age;
    double height;

    cout << "Enter your name: ";
    cin >> name;
    cout << "Enter your age: ";
    cin >> age;
    cout << "Enter your height: ";
    cin >> height;

    //Print a blank line
    cout << endl;

    //Show the details you typed
    cout << "Name is " << name << endl;
    cout << "Age is " << age << endl;
    cout << "Height is " << height << endl;
}
```

Enter your name: John
 Enter your age: 18
 Enter your height: 160.5

Name is John.
 Age is 18.
 Height is 160.5.

مثال ۱-۹ برنامه‌ای که یک متغیر صحیح و یک متغیر اعشاری را اعلان کرده، مقادیری به آنها نسبت می‌دهد و سپس آنها را به خروجی می‌برد.

۱. برنامه‌ی زیر را در فایل ۱-۹.cpp تایپ کنید:

```
#include "stdafx.h"
#include <iostream>
int main()
{
    int x = 10;
    float y = 5.5;
    std::cout << "x = " << x;
    std::cout << " y = " << y;
    std::cin.get();
}
```

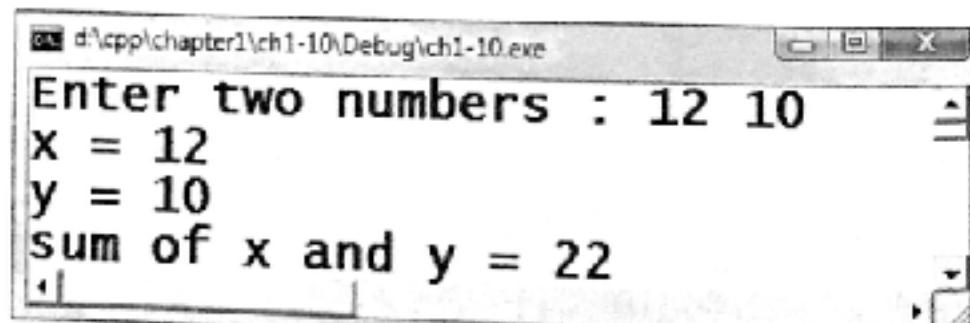
۲. برنامه را اجرا کنید تا خروجی آن را به صورت زیر ببینید:

مثال ۱-۱۰ برنامه‌ای که دو مقدار صحیح را از ورودی می‌خواند و مجموع آن دو مقدار را چاپ می‌کند.

۱. برنامه‌ی زیر را در فایل ch1-10.cpp تایپ کنید:

```
#include "stdafx.h"
#include <iostream>
using namespace std;
int main()
{
    int x, y;
    cout << "Enter two numbers : ";
    cin >> x >> y ;
    cout << "x = " << x << endl;
    cout << "y = " << y << "\n";
    cout << "sum of x and y = " << x + y;
    cin.ignore(); // ignore end of line char
    cin.get();
}
```

۲. برنامه را اجرا کنید تا خروجی آن را به صورت زیر بینید:



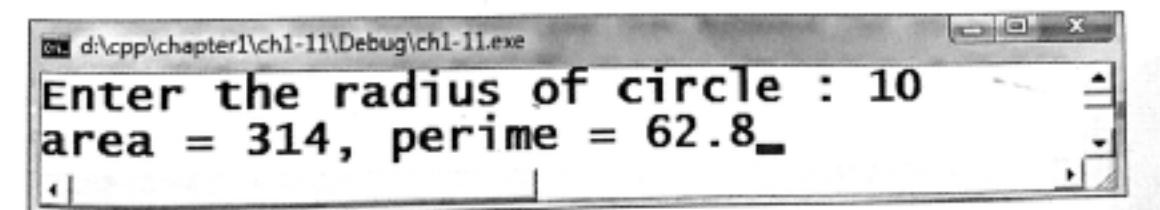
مثال ۱-۱۱ برنامه‌ای که شعاع دایره‌ای را که یک مقدار صحیح است از ورودی می‌خواند و مساحت و محیط دایره را محاسبه می‌کند و به خروجی می‌برد. در این برنامه سه متغیر به کار رفته‌اند که وظایف آن‌ها عبارتند از:

- شعاع دایره : radius
- مساحت دایره : area
- محیط دایره : perime

۱. برنامه‌ی زیر را در فایل ch1-11.cpp تایپ کنید:

```
#include "stdafx.h"
#include <iostream>
#define PI 3.14
using namespace std;
int main()
{
    int radius;
    float area, perime;
    cout << "Enter the radius of circle : ";
    cin >> radius;
    area = PI * radius * radius;
    perime = 2 * PI * radius;
    cout << "area = " << area << ", perime = " << perime;
    cin.get(); // delete end of line char
    cin.get();
}
```

۲. برنامه را اجرا کنید تا خروجی آن را به صورت زیر ببینید:



- مثال ۱-۱۲** برنامه‌ای که سه مقدار صحیح را از صفحه کلید خوانده می‌انگین آن‌ها را محاسبه می‌کند و به خروجی می‌برد (به شرح مثال توجه کنید).
۱. برنامه‌ی زیر را در فایل ۱-۱۲.cpp تایپ کنید:

```
#include "stdafx.h"
#include <iostream>
using namespace std;
int main()
{
    int x, y, m;
    float ave;
    cout << "Enter three integer numbers : ";
    cin >> x >> y >> m;
    ave = static_cast<float> (x + y + m) / 3;
    cout << "ave = " << ave;
    cin.ignore();
    cin.get();
}
```

۲. برنامه را اجرا کنید تا خروجی آن را به صورت زیر ببینید:

