

یادآوری: تمامی تمرینات و اطلاعات مربوط به تحویل آنها در سایت درس قرار داده میشوند:

<http://ele.aut.ac.ir/~btaheri/cpp/>

## توابع کتابخانه‌های:

در بسیاری از موارد، با استفاده از توابع آماده که در کتابخانه‌های C++ تعریف شده‌اند، نیازی به تعریف آنها توسط خود کاربر نیست. در اینجا چند نمونه از این توابع را می‌بینیم:

### توابع کتابخانه ریاضی (cmath):

توابع مثلثاتی:

به کد زیر دقت کنید:

```
double tanh (double x);  
float tanh (float x);  
long double tanh (long double x);
```

کد فوق را در برنامه‌های که مقادیر  $\tanh$  آرگومانهای ورودی را بازگرداند، تکمیل کنید و توضیح دهید که آیا امکان دارد که چند تابع همانام را بصورت فوق در برنامه قرار داد؟ چگونه؟

تحقیق کنید که جنس آرگومانهای ورودی در توابع مثلثاتی  $\sin()$ ،  $\cos()$ ،  $\tan()$  چه باید باشد؟ جنس مقدار بازگشتی توابع معکوس مثلثاتی چطور؟

### توابع کتابخانه ریاضی (iomanip):

توابع موجود در این کتابخانه غالباً برای تنظیم نحوه چاپ خروجیها (بهویژه داده‌های عددی) روی صفحه نمایش کاربرد دارند.

به برنامه زیر توجه کنید:

```
1. //File:SetW.cpp  
2. #include <iostream.h>  
3. #include <iomanip.h>  
4. int main ()  
5. {  
6.     double x=4.25,y=6;  
7.     cout << setw(10) << x << endl;  
8.     cout << setw(10) << y;  
9.     return 0;  
10.}
```

آن را اجرا کنید و توضیح دهید که تابع  $\text{setw}()$  به چه ترتیب کار میکند.

مثال:

برنامه زیر را اجرا کنید و باتوجه به خروجی چاپ شده کارکرد تابع `setfill()` را توضیح دهید:

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

int main ()
{
    cout << setfill ('x') << setw (10);
    cout << 77 << endl;
    return 0;
}
```

مثال:

با در نظر گرفتن کد زیر، توضیح دهید که چگونه میتوان نحوه نمایش بخش اعشاری در اعداد اعشاری را کنترل نمود:

```
1. //File:ShowPoint.cpp
2. #include <iostream.h> //or #include <iomanip.h>
3. using namespace std;
4. int main()
5. {
6.     float a,b,c;
7.     a=4;
8.     b=6;
9.     c=2.75;
10.    cout << "a=" << a << "b=" << b << "c=" << c << endl;
11.    cout << showpoint << "a=" << a << "b=" << b << "c=" << c;
12.    return 0;
13. }
```

## تولید اعداد تصادفی:

با استفاده از توابع موجود در کتابخانه `cstdlib` میتوان اعداد تصادفی تولید نمود.

مثال:

برنامه زیر را در نظر بگیرید:

```
1. //File:Rand.cpp
2. #include <iostream>
3. #include <iomanip>
4. #include <cstdlib> // Need this for the rand() function
5. int main(void)
6. {
7.     cout << "RAND_MAX = " << RAND_MAX << "\n\n";
8.     cout << "The first 10 random numbers are...\n";
9.     for(int i = 1; i <= 10; i = i + 1)
10.        cout << "Random number "<< setw(2)<< i << " = "
11.        << rand() // Random int between 0 and RAND_MAX
12.        << "\n";
}
```

```
13.     return 0;
14. }
```

سوالات:

۱. مقدار بازگردانده توسط `RAND_MAX` در خط ۷ بیانگر چه می باشد؟
۲. توضیح دهید که حلقه قرار گرفته در سطر ۹-۱۲ چه کار میکند؟
۳. دلیل قرار دادن تابع `setw(2)` چه می باشد؟
۴. برنامه را مجدداً اجرا کنید. درمورد نحوه عملکرد تابع `rand()` چه نتیجه‌ای می‌گیرید؟

مثال:

برنامه زیر را در نظر بگیرید:

```
1. //File:Srand2.cpp
2. #include <iostream>
3. #include <iomanip>
4. #include <cstdlib> // Need this for the rand()
5. #include <ctime> // Need this for the time() function
6. int main(void)
7. {
8.     cout << "RAND_MAX = " << RAND_MAX << "\n\n";
9.     srand(time(NULL)); //Seed the random number generator
10.    cout << "The first 10 random numbers are...\n";
11.    for(int i = 1; i <= 10; i = i + 1)
12.        cout << "Random number "
13.              << setw(2)
14.              << i << " = "
15.              << rand() // Random int between 0 and RAND_MAX
16.              << "\n";
17.    return 0;
18. }
```

سوالات:

۱. تفاوت تابع `rand()` با تابع `srand()` بکاررفته در خط ۹ از برنامه فوق را توضیح دهید.
۲. دلیل استفاده از تابع `time(NULL)` در آرگومان تابع `srand()` چه می باشد؟
۳. تابع `time(NULL)` بازگردانده چه می باشد؟
۴. توضیح دهید که برنامه کنونی نسبت به برنامه مثال قبل چه مزیتی دارد.
۵. برنامه فوق را به گونه‌ای تغییر دهید که اعداد تصادفی را در بازه `[0,1]` تولید کند.
۶. برنامه فوق را به گونه‌ای تغییر دهید که اعداد تصادفی را در بازه `[0,15]` تولید کند.

## آرایه‌ها:

آرایه‌ها ابزاری برای ذخیره‌سازی داده‌ها هستند که امکان دریافت و ذخیره نمودن تعداد زیادی داده را فراهم می‌کنند. درواقع آرایه‌ها ساختاری مشابه ماتریسها دارند که هریک از درایه‌های آنها دارای مختصات مشخصی می‌باشد.

مثال:

برنامه زیر را در نظر بگیرید:

```

1. //File:SinArray.cpp
2. #include <iostream>
3. #include <math.h>
4. using namespace std;
5.
6. int main()
7. {
8.     int x[10];
9.     double y[10];
10.    for (int i=0;i<10;i++)
11.    {
12.        x[i]=i;
13.        y[i]=sin(x[i]);
14.    }
15.    for (int counter=0;counter<10;counter++)
16.    {
17.        cout<<y[counter]<<"\t";
18.    }
19.    return 0;
20. }

```

سوالات:

۱. به نحوه تعریف آرایه‌های X و Y در سطرهای ۸ و ۹ برنامه توجه کنید. عدد ۱۰ در اندیس این آرایه‌ها بیانگر چه می‌باشد؟
۲. با توجه به حلقه موجود در سطرهای ۱۴-۱۰ نحوه مقداردهی به یک آرایه را توضیح دهید.
۳. خروجی برنامه فوق را پیش‌بینی کنید.

آرایه‌های چندبعدی:

مشابه با ماتریس‌های چندبعدی، امکان ایجاد آرایه‌های چندبعدی نیز وجود دارد.

مثال:

کد زیر را در نظر بگیرید:

```

const int NROWS = 3;
const int NCOLUMNS = 2;
int a[NROWS][NCOLUMNS] = {0};

```

آن را تکمیل کنید و یک روش از مقداردهی به آرایه‌ها را توضیح دهید.

مثال:

کد زیر را تکمیل کنید و به سوالات زیر پاسخ دهید:

```

const int NROWS = 3;
const int NCOLUMNS = 2;
int a[NROWS][NCOLUMNS] =
{
    {1,2},
    {3,4},
    {5,6}
};

```

سوالات:

۱. آیا می‌توان سائز یک آرایه را با یک مقدار متغیر تعیین کرد؟ درستی ادعای خود را تحقیق کنید.
۲. در کد فوق چنانچه یک زوج از درایه‌های را مقداردهی نکنیم، انتظار دارید در خروجی چه مشاهده کنید.