

### const pointer to const

```
int main()
{
    int a=10,b=20;

    const int * const p=&a; ✓
    a=15; ✓
    *p=18; ✗
    (p=&b); ✗
    b=25; ✓
    *p=30; ✗
    return 0;
}
```

### const function argument

```
#include <iostream>
using namespace std;
int mystrlen(const char *p)
{
    int i;
    for(i=0;*(p+i)!='\0';i++);
    (*p)++;
    return i;
}
int main()
{
    char city[]={"BHOPAL"};
    int x;
    x=mystrlen(city);
    cout<<"City is "<<city<<endl;
    cout<<"It's length is "<<x;
    return 0;
}
```

This code will stop  
us from changing  
bp !

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

### const data members

```
{
    int rad;
    const float pi=3.14;
public:
    Circle(int r)
    {
        rad=r;
    }
    void area()
    {
        float a;
        a=++pi*rad*rad;
        cout<<"area is "<<a;
    }
};
int main()
{
    Circle obj(10);
    obj.area();
    return 0;
}
```

Since pi is const we must initialize it at the decl. point

This code will give error because we are changing pi

### Another Way Of Initializing const Data

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

```
{
    const int roll;
    int marks;
    float per;
public:
    Student(int r,int m,float p): roll(r),marks(m),per(p)
    {
    }
    void show()
    {
        cout<<"Roll="<<roll<<"<Marks="<<marks<<"<Per="<<per<<endl;
    }
};
int main()
{
    Student S1(101,60,65);
    Student S2(102,70,75);
    S1.show();
    S2.show();
    return 0;
}
```

This is called initializer list!