

## Pr13-12.cpp

```
// This program demonstrates a class with a destructor.
#include <iostream>
#include "ContactInfo.h"
using namespace std;

int main()
{
    ContactInfo entry("Kristen Lee", "555-2021", 25);
    ContactInfo blank;

    cout << "Name: " << entry.getName() << endl;
    cout << "Phone Number: " << entry.getPhoneNumber() << endl;
    cout << "Age: " << entry.getAge() << endl;

    cout << endl;

    cout << "Name: " << blank.getName() << endl;
    cout << "Phone Number: " << blank.getPhoneNumber() << endl;
    cout << "Age: " << blank.getAge() << endl;

    return 0;
}
```

## ContactInfo.h

```
// Contact class specification file (version 2)
#ifndef CONTACTINFO_H
#define CONTACTINFO_H
#include <string> // Needed for strlen and strcpy

using namespace std;

// ContactInfo class declaration.
```

```
class ContactInfo
{
private:
    string name; // The contact's name
    string phone; // The contact's phone number
    int age;

    void initName(string n);

    void initPhone(string p);

    void initAge(int a);

public:
    //Default Constructor
    ContactInfo()
    {
        initName("James bond");
        initPhone("555-555");
        initAge(0);
    }

    // Constructor
    ContactInfo(string n, string p, int a)
    {
        // Initialize the name attribute.
        initName(n);

        // Initialize the phone attribute.
        initPhone(n);

        //Initialize the age
        initAge(a);
    }
}
```

```
    }

    // Destructor
    ~ContactInfo()
    {
    }

    const string getName();

    const string getPhoneNumber();

    const int getAge();
};
#endif
```

contactInfo.cpp

```
#include "ContactInfo.h"
#include <iostream>
#include <iomanip>

using namespace std;

void ContactInfo::initName(string n)
{
    name = n;
}

void ContactInfo::initPhone(string p)
{
    phone = p;
}

void ContactInfo::initAge(int a)
```

```
{
    age = a;
}

const string ContactInfo::getName()
{
    return name;
}

const string ContactInfo::getPhoneNumber()
{
    return phone;
}

const int ContactInfo::getAge()
{
    return age;
}
```

Output:

```
Name: Kristen Lee
Phone Number: Kristen Lee
Age: 25

Name: James bond
Phone Number: 555-555
Age: 0
```