Programa que pide una fecha al usuario y emplea el operador de asignar el valor de una
instancia, hace otras dos. Realiza una asignación múltiple de instancias. La función
miembro retorna el apuntador this.

```cpp
#ifndef CONTACTINFO_H
#define CONTACTINFO_H
#include <cstring> // Needed for strlen and strcpy
// ContactInfo class declaration.
class ContactInfo
{
private:
       char *name; // The contact's name
       char *phone; // The contact's phone number
public:
       // Constructor
       ContactInfo(char *n, char *p);
       ContactInfo();
       // Destructor
       ~ContactInfo();
       // This function initializes the name attribute.
       void setName(char *n);
       // This function initializes the phone attribute.
       void setPhone(char *p);
       const char *getName() const;
       const char *getPhoneNumber() const;
       void display() const;
       const ContactInfo &operator=(const ContactInfo &obj);
#endif
       +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
#include<iostream>
using namespace::std;
#include "ContactInfo.h"
       ContactInfo::ContactInfo(){
              char tempName[20] = "Jose Rivera";
              char tempPhone[20] = "000-000-0000";
              setName(tempName);
              setPhone(tempPhone);
       }
       //Constructor
       ContactInfo::ContactInfo(char *n, char *p)
       { // Initialize the name attribute.
              setName(n);
              // Initialize the phone attribute.
              setPhone(p);
       }
       // Destructor
       ContactInfo::~ContactInfo()
       {
              delete[] name;
              delete[] phone;
       }
       void ContactInfo::setName(char *n)
       {
              name = new char[strlen(n) + 1];
              strcpy_s(name, strlen(n) + 1, n);
       }
```

```cpp
// Private member function: initPhone
// This function initializes the phone attribute.
void ContactInfo::setPhone(char *p)
{
        phone = new char[strlen(p) + 1];
        strcpy_s(phone, strlen(p) + 1, p);
}
const char *ContactInfo::getName() const
{
        return name;
}
const char *ContactInfo::getPhoneNumber() const
{
        return phone;
}
void ContactInfo::display() const{
        cout << "Nombre:" << getName() << endl;
        cout << "Telephono:" << getPhoneNumber() << endl;
};
const ContactInfo &ContactInfo::operator=(const ContactInfo &obj){
        delete[] name;
        name = new char[strlen(obj.getName()) + 1];
        strcpy_s(name, strlen(obj.getName()) + 1, obj.getName());
        delete[] phone;
        phone = new char[strlen(obj.getPhoneNumber()) + 1];
        strcpy_s(phone, strlen(obj.getPhoneNumber()) + 1, obj.getPhoneNumber());
        return *this;
}
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ +
#include<iostream>
#include<string>
        using namespace::std;
#include "ContactInfo.h"
    int main(){
        ContactInfo std1, std2, std3;
        char *name = nullptr;
        char *phoneNumber = nullptr;
        const int SIZE = 15;
        name = new char[SIZE];
        phoneNumber = new char[SIZE];
        cout << "Entre su nombre:";
        cin.getline(name, SIZE);
        cout << "Entre su numero de telefono:";
        cin.getline(phoneNumber, SIZE);
        std1.setName(name);
        std1.setPhone(phoneNumber);
        std3 = std2 = std1;
        std1.display();
        std2.display();
        std3.display();
        delete[] name;
        name = nullptr;
        delete[] phoneNumber;
        phoneNumber = nullptr;
        system("pause");
        return 0;
    }//end main
```