

THE HONG KONG POLYTECHNIC UNIVERSITY
Department of Electronic and Information Engineering

Computer Programming (ENG236) - Homework 5

- A. By using Visual Studio 2005, develop a static library, namely `Cards.lib`, based on the class specifications given in `Cards.h` below. To do this part, you may prepare your own card picture files or download the file `Cards.zip` from the subject Web page. Store the card picture files to a folder `cards_gif` in your project folder.

```
// Implement using visual studio 2005 a static library.
// The classes Card and CardFrame are defined below.
// The class Card has been implemented. You are required to implement the
// member functions of CardFrame
// (Words started with $ in the comments refer to the member variables as stated
// in the private sections)

class Card
{
public:
    int getsuit() {return suit;}
    // Return $suit
    int getnumber() {return number;}
    // Return $number
    char * getfilename() {return filename;}
    // Return the character array $filename by a pointer
    void setsuit(int s) {suit = s;}
    // $suit = s
    void setnumber(int n) {number = n;}
    // $number = n
    void setfilename(char * fn) {strncpy(filename,fn,100);}
    // Copy fn[] to $filename[]

private:
    int suit;           // 1 = spade, 2 = heart, 3 = club, 4 = diamond
    int number;        // Represent the number of the cards.
                        // number = 2 to 10 for card no = 2 to 10
                        // Jack = 11, Queen = 12, King = 13, Ace = 1
    char filename[100]; // The filename of the card
};

class CardFrame
{
public:
    CardFrame(int num);
    // When an object of this class is instantiated, do the following
    // $cardNum = num
    // Create an array of num objects of the class Card in the heap.
    // The pointer of the array should be saved in $pCardArr

    ~CardFrame();
    // Delete all Card in the heap

    void genCard(int array_index);
    // When this function is called, do the following
    // 1. Generate a random number in the range 1 to 4 and save into $suit of
    // the "array_index" Card in the array. E.g. if array_index = 0, the
    // suit of the first Card in the array should be randomly generated
    // and saved into its $suit variable.
    // 2. Generate a random number in the range 1 to 13 and save into $number
    // of the "array_index" Card in the array. E.g. if array_index = 0,
    // the card number of the first Card in the array should be randomly
    // generated and saved into its $number variable.

    char * genFilename(int array_index);
    // When this function is called, construct the filename of the
```

```

// "array_index" Card based on the values of $suit and $number and save
// into $filename[]. E.g., assume your cards_gif folder is stored in
// e:\HW5, $suit and $number of the "array_index" Card are 2 and 12,
// respectively. The filename you should generate is
// "e:\\HW5\\cards_gif\\hq.gif" (i.e. Heart Queen)
// Note that \ is a special character. It has to be preceded by another \
// in a string.
// Hint: Use strcat to combine the strings together and form a filename
// Return $filename of the "array_index" Card

```

```

private:
    int cardNum;           // Keep the number of cards
    Card * pCardArr;      // Keep the pointer that points to an array of
                          // Card in heap
};

```

Write a console application to test all the member functions of `CardFrame` using the developed static library.

B. By using Visual Studio 2005, develop a form with specifications as defined below:

1. You are required to make use of the library you developed in Question A in this Form project.
2. When the program is executed, the form in Fig.1 should be shown. In the form, 3 `pictureBox`'s are created and show 3 cards with face down.
3. Whenever the user clicks on any of the covered cards, a card of random suit and number will be shown as in Fig.2.
4. If the user clicks on any of the uncovered cards, the card will become covered again. Repeat step 3 to ensure that the covered card can be clicked again to show another card of random suit and number.

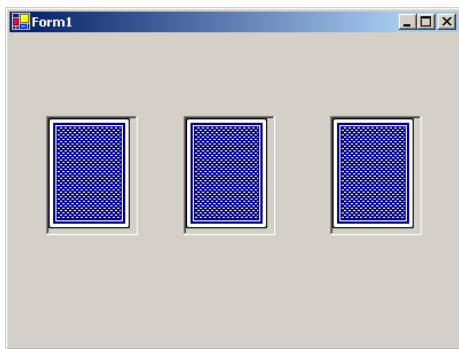


Fig.1

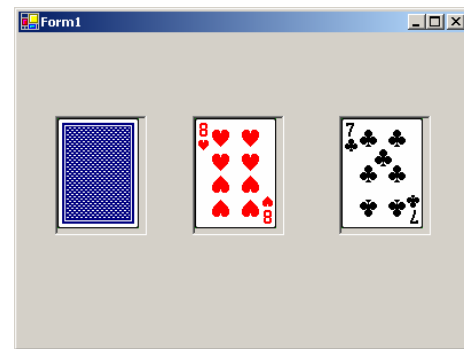
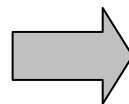


Fig.2

Hint 1: Add a private member variable with the following statement `"CardFrame *pCardFrame;"` in `Form1.h` (remember to include `Cards.h` at the beginning). In the constructor of `Form1.h`, create a `CardFrame` object in the heap. Save the pointer to `pCardFrame`.

Hint 2: If a character array `fn[]` contains the filename of a picture as follows:

```
char fn[] = "e:\\HW5\\cards_gif\\hq.gif";
```

you can display the picture in a `pictureBox`, say, `pictureBox1`, by

```
pictureBox1->Image = Image::FromFile(fn);
```

Instructions

You need to follow exactly the requirement of the questions as shown above when developing your programs. Failure to do so is unacceptable even if your programs can give a similar result. Try to explain your program as clear as possible using comments.

```

// The following are in the file "Cards.h" under the project "HW5Lib"
#include <string>
class Card
{
public:
    int getsuit() {return suit;}
    // Return $suit
    int getnumber() {return number;}
    // Return $number
    char * getfilename() {return filename;}
    // Return the character array $filename by a pointer
    void setsuit(int s) {suit = s;}
    // $suit = s
    void setnumber(int n) {number = n;}
    // $number = n
    void setfilename(char * fn) {strncpy(filename,fn,100);}
    // Copy fn[] to $filename[]
private:
    int suit;           // 1 = spade, 2 = heart, 3 = club, 4 = diamond
    int number;        // Represent the number of the cards.
                       // number = 2 to 10 for card no = 2 to 10
                       // Jack = 11, Queen = 12, King = 13, Ace = 1
    char filename[100]; // The filename of the card
};

class CardFrame
{
public:
    CardFrame(int num);
    // When the object is instantiated, do the following
    //   $cardNum = num
    //   Create an array of num objects of the class Card in the heap.
    //   The pointer of the array should be saved in $pCardArr

    ~CardFrame();
    // Delete all Card in the heap

    void genCard(int array_index);
    // When this function is called, do the following
    // 1. Generate a random number in the range 1 to 4 and save into $suit of
    //    the "array_index" Card in the array. E.g. if array_index = 0, the
    //    suit of the first Card in the array should be generated and saved
    //    into its $suit variable.
    // 2. Generate a random number in the range 1 to 13 and save into $number
    //    of the "array_index" Card in the array. E.g. if array_index = 0,
    //    the card number of the first Card in the array should be generated
    //    and saved into its $number variable.

    char * genFilename(int array_index);
    // When this function is called, construct the filename of the
    // "array_index" Card based on the values of $suit and $number and save
    // into $filename[]. E.g., assume your cards_gif folder is stored in
    // c:\HW4, $suit and $number of the "array_index" Card are 2 and 12,
    // respectively. The filename you should generate is
    // "c:\HW4\cards_gif\hq.gif" (i.e. Heart Queen)
    // Note that \ is a special character. It has to be preceded by another \
    // in a string.
    // Hint: Use strcat to combine the strings together and form a filename
    // Return $filename of the "array_index" Card

private:
    int cardNum;           // Keep the number of cards
    Card * pCardArr;      // Keep the pointer that points to an array of Card in heap
};

```

```

// The following are in the file "Cards_code.cpp" under the project "HW5Lib"
#include "Cards.h"
#include <stdlib.h>
#include <time.h>
CardFrame::CardFrame(int num)
{
    cardNum = num;
    pCardArr = new Card[num];
    srand(time(NULL)); //Ensure every running has different random cards generated.
}

```

```

CardFrame::~CardFrame()
{
    delete [] pCardArr; //Delete all Card in the heap
}

void CardFrame::genCard(int array_index)
{
    int suit, number;

    suit = rand() % 4 + 1; //generate a random in the range 1 to 4
    number = rand() % 13 + 1; //generate a random in the range 1 to 13
    pCardArr[array_index].setsuit(suit);
    pCardArr[array_index].setnumber(number);
}

char * CardFrame::genFilename(int array_index)
{
    int suit = pCardArr[array_index].getsuit();
    int number = pCardArr[array_index].getnumber();
    char fileName[100] = "e:\\Temp\\Cards\\";
    char abc[10]; // storing words in the filename

    switch (suit)
    {
        case 1:
            strcat(fileName, "s");
            break;
        case 2:
            strcat(fileName, "h");
            break;
        case 3:
            strcat(fileName, "c");
            break;
        case 4:
            strcat(fileName, "d");
            break;
    }
    switch(number)
    {
        case 11:
            strcat(fileName, "j");
            break;
        case 12:
            strcat(fileName, "q");
            break;
        case 13:
            strcat(fileName, "k");
            break;
        default:
            itoa(number, abc, 10); // Convert integer to string
            strcat(fileName, abc);
            break;
    }
    strcat(fileName, ".gif"); // Whole name stored in fileName
    pCardArr[array_index].setfilename(fileName);
    return pCardArr[array_index].getfilename();
}

```

```

// The following are in the file "TestMain.cpp" under project "HW5TMain", with "Cards.h"
// and "HW5Lib.lib" in the same folder. Test all the member functions of CardFrame
// To make the program below work, the CLR SUPPORT MUST BE DISABLED when building the
// static library

```

```

#include "Cards.h"
#include <iostream>
using namespace std;

int main()
{
    CardFrame *cf;
    cf=new CardFrame(3);

    cf->genCard(0); //Generate Card0
    cout<<"The returned filename is: " <<cf->genFilename(0)<<endl;

    cf->genCard(1); //Generate Card1
    cout<<"The returned filename is: " <<cf->genFilename(1)<<endl;
}

```

```

    cf->genCard(2); //Generate Card2
    cout<<"The returned filename is: " <<cf->genFilename(2)<<endl;
    return 0;
}

// The following are in the file "Form1.h" under the project name "HW5Form",
// with "Cards.h" and "HW5Lib.lib" in the same folder.
// To make the program below work, the CLR SUPPORT MUST BE ENABLED when building the
// static library
// Suggested solution for Part B
#pragma once
#include "Cards.h"

namespace HW5Form {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    using namespace System::Runtime::InteropServices;

    /// <summary>
    /// Summary for Form1
    ///
    /// WARNING: If you change the name of this class, you will need to change the
    /// 'Resource File Name' property for the managed resource compiler tool
    /// associated with all .resx files this class depends on. Otherwise,
    /// the designers will not be able to interact properly with localized
    /// resources associated with this form.
    /// </summary>
    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            cf=new CardFrame(3); //Create objects of the class CardFrame//
            showingBack1=true; //Initially cards are covered
            showingBack2=true;
            showingBack3=true;
        }

    protected:
        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::PictureBox^ pictureBox1;
    protected:
    private: System::Windows::Forms::PictureBox^ pictureBox2;
    private: System::Windows::Forms::PictureBox^ pictureBox3;

    private:

        CardFrame *cf; // Create an array of Cards in the heap
        bool showingBack1, showingBack2, showingBack3;

        System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        void InitializeComponent(void)
        {

```

```

        System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(Form1::typeid));
        this->pictureBox1 = (gcnew System::Windows::Forms::PictureBox());
        this->pictureBox2 = (gcnew System::Windows::Forms::PictureBox());
        this->pictureBox3 = (gcnew System::Windows::Forms::PictureBox());
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox1))->BeginInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox2))->BeginInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox3))->BeginInit();
        this->SuspendLayout();
        //
        // pictureBox1
        //
        this->pictureBox1->Image = (cli::safe_cast<System::Drawing::Image^
>(resources->GetObject(L"pictureBox1.Image")));
        this->pictureBox1->Location = System::Drawing::Point(38, 34);
        this->pictureBox1->Name = L"pictureBox1";
        this->pictureBox1->Size = System::Drawing::Size(103, 138);
        this->pictureBox1->TabIndex = 0;
        this->pictureBox1->TabStop = false;
        this->pictureBox1->Click += gcnew System::EventHandler(this,
&Form1::pictureBox1_Click);
        //
        // pictureBox2
        //
        this->pictureBox2->Image = (cli::safe_cast<System::Drawing::Image^
>(resources->GetObject(L"pictureBox2.Image")));
        this->pictureBox2->Location = System::Drawing::Point(174, 34);
        this->pictureBox2->Name = L"pictureBox2";
        this->pictureBox2->Size = System::Drawing::Size(100, 138);
        this->pictureBox2->TabIndex = 1;
        this->pictureBox2->TabStop = false;
        this->pictureBox2->Click += gcnew System::EventHandler(this,
&Form1::pictureBox2_Click);
        //
        // pictureBox3
        //
        this->pictureBox3->Image = (cli::safe_cast<System::Drawing::Image^
>(resources->GetObject(L"pictureBox3.Image")));
        this->pictureBox3->Location = System::Drawing::Point(307, 34);
        this->pictureBox3->Name = L"pictureBox3";
        this->pictureBox3->Size = System::Drawing::Size(100, 138);
        this->pictureBox3->TabIndex = 2;
        this->pictureBox3->TabStop = false;
        this->pictureBox3->Click += gcnew System::EventHandler(this,
&Form1::pictureBox3_Click);
        //
        // Form1
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(449, 273);
        this->Controls->Add(this->pictureBox3);
        this->Controls->Add(this->pictureBox2);
        this->Controls->Add(this->pictureBox1);
        this->Name = L"Form1";
        this->Text = L"Form1";
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox1))->EndInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox2))->EndInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^ >(this-
>pictureBox3))->EndInit();
        this->ResumeLayout(false);

    }
#pragma endregion
    private: System::Void pictureBox1_Click(System::Object^ sender, System::EventArgs^
e) {
        if (showingBack1) // Card is covered, show the card
        {
            cf->genCard(0);
            String ^ CardStr=Marshal::PtrToStringAnsi((IntPtr)cf-
>genFilename(0));
            pictureBox1->Image=Image::FromFile(CardStr);

```

```

    }
    else // Card is uncovered, cover it
        pictureBox1->Image=Image::FromFile("e:\\Temp\\Cards\\blfv.gif");
        showingBack1=!showingBack1; // Toggle the flag
    }
}

private: System::Void pictureBox2_Click(System::Object^ sender, System::EventArgs^
e) {
    if (showingBack2) // Card is covered, show the card
    {
        cf->genCard(0);
        String ^ CardStr=Marshal::PtrToStringAnsi((IntPtr)cf-
>genFilename(0));
        pictureBox2->Image=Image::FromFile(CardStr);
    }
    else // Card is uncovered, cover it
        pictureBox2->Image=Image::FromFile("e:\\Temp\\Cards\\blfv.gif");
        showingBack2=!showingBack2; // Toggle the flag
    }
}

private: System::Void pictureBox3_Click(System::Object^ sender, System::EventArgs^
e) {
    if (showingBack3) // Card is covered, show the card
    {
        cf->genCard(0);
        String ^ CardStr=Marshal::PtrToStringAnsi((IntPtr)cf-
>genFilename(0));
        pictureBox3->Image=Image::FromFile(CardStr);
    }
    else // Card is uncovered, cover it
        pictureBox3->Image=Image::FromFile("e:\\Temp\\Cards\\blfv.gif");
        showingBack3=!showingBack3; // Toggle the flag
    }
};
}

```