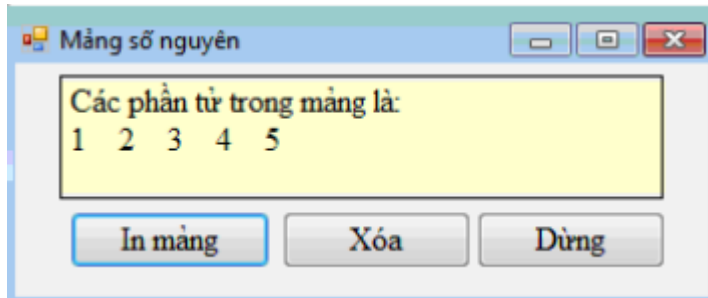


Bài 1: Hãy thiết kế một ứng dụng Windows Forms Application như hình bên dưới:



* Yêu cầu

- Thiết kế form như mẫu (lblKQ, btnIn, btnXoa, btnDung).
- Khai báo mảng 1 chiều (a) gồm 5 số nguyên từ 1 đến 5.
- Nhấp vào button **In mảng** (btnIn) sẽ in ra label (lblKQ) các giá trị trong mảng.
- Nhấp vào button **Xóa** (btnXoa) sẽ xóa trống nội dung của label (lblKQ).
- Nhấp vào button **Dừng** sẽ dừng chương trình.

* Hướng dẫn

- Thiết kế form như yêu cầu.
- Khai báo mảng: qua code, thêm đoạn code để được kết quả như sau:

```
public partial class Form1 : Form
{
    // Khai báo mảng 1 chiều gồm 5 số nguyên từ 1 đến 5
    int[] a = { 1, 2, 3, 4, 5 };
}
```

- Nhấp đúp vào button **In mảng**, thêm đoạn code sau:

```
// Xuất các phần tử trong mảng ra màn hình
this.lblKQ.Text="Các phần tử trong mảng là:\n\r";
for (int i=0; i < a.Length; i++) {
    this.lblKQ.Text += a[i]+"    ";
}
```

Có thể thay for bằng foreach như sau

```
foreach (int phantu in a) {
```

Bài 2: Hãy thiết kế một ứng dụng Windows Forms Application như hình bên dưới:



* Yêu cầu

- Thiết kế form: btnNhap, txtNhap, btnTang, btnGiam, lblKQ, btnIn, btnXoa, btnDung.
- Khai báo mảng 1 chiều (a) chứa 5 số nguyên, số phần tử hiện có (sopt) là 0.
- Nhập số vào TextBox txtNhap, nhấn vào button **Nhập 1 phần tử mảng** (btnNhap) cho phép đưa giá trị trong TextBox txtNhap vào mảng. Khi mảng đã đủ 5 số nguyên thì phải thông báo “Mảng đã đầy” và không cho nhập nữa.
- Nhấn vào button **Sắp tăng** (btnTang) sẽ sắp xếp mảng theo thứ tự tăng dần.
- Nhấn vào button **Sắp giảm** (btnGiam) sẽ sắp xếp mảng theo thứ tự giảm dần.
- Nhấn vào button **In mảng** (btnIn) sẽ in ra label (lblKQ) các giá trị trong mảng.
- Nhấn vào button **Xóa** (btnXoa) sẽ xóa trống nội dung của label (lblKQ) đồng thời khai báo lại số phần tử hiện có (sopt) của mảng là 0.
- Nhấn vào button **Dừng** sẽ dừng chương trình.

* Hướng dẫn

- Thiết kế form như yêu cầu.
- Khai báo mảng như sau:

```
public partial class Form1 : Form
{
    // Khai báo mảng 1 chiều gom 5 số nguyên
    int[] a = new int[5];
    // Khai báo số phần tử hiện có của mảng
    int sopt = 0;
}
```

- Nhấp đúp vào button **Nhập 1 phần tử mảng**, thêm đoạn code sau:

```
// Nhập một phần tử cho mảng
if (sopt == 5)
    MessageBox.Show("Mảng đã đầy!");
else
{
    a[sopt] = Convert.ToInt32(this.txtNhap.Text);
    sopt++;
    this.txtNhap.ResetText();
    this.txtNhap.Focus();
}
```

- Nhấp đúp vào button **Sắp tăng**, thêm đoạn code sau:

```
// Sắp xếp mảng giảm
if (sopt == 0)
    this.lblKQ.Text = "Mảng rỗng!";

else
    Array.Sort(a, 0, sopt);
this.lblKQ.Text = "Đã sắp xếp mảng tăng dần!";
```

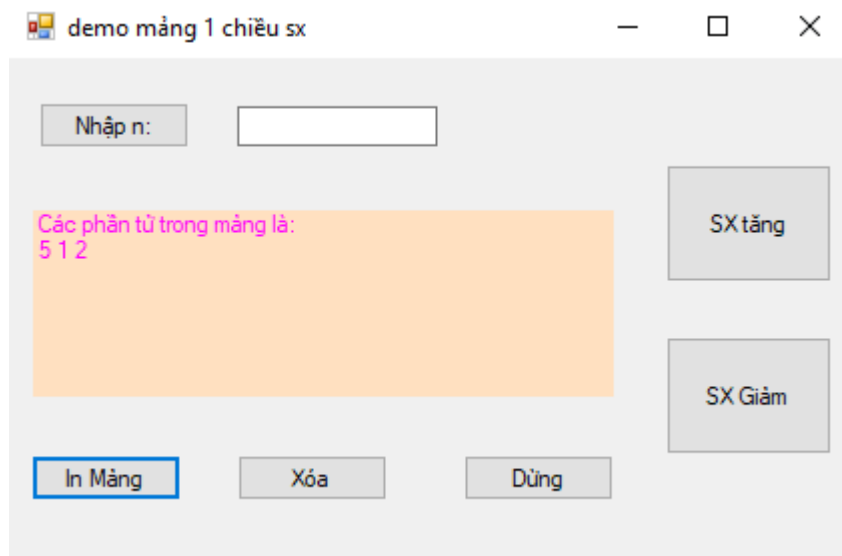
// sv làm tương tự cho nút sx giảm

```
Array.Reverse(a, 0, sopt);
```

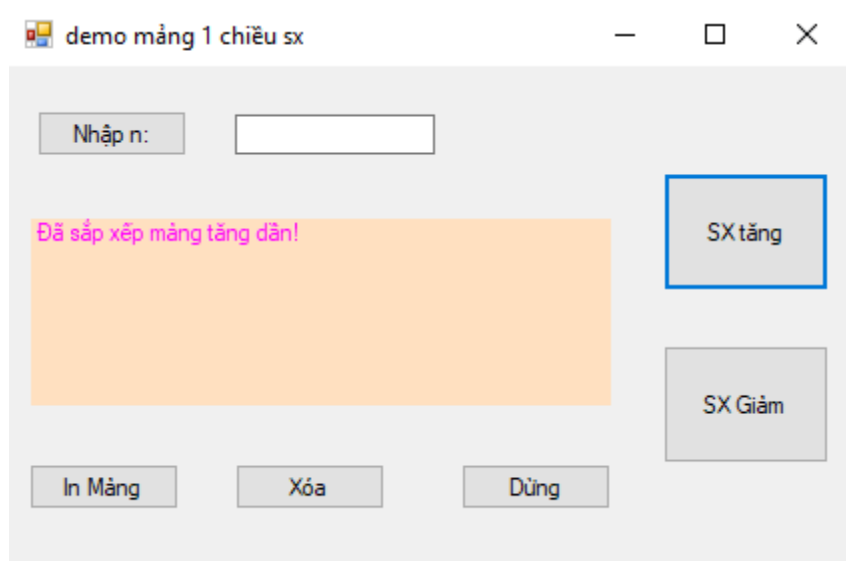
Nhập n lần lượt là

5,1,2

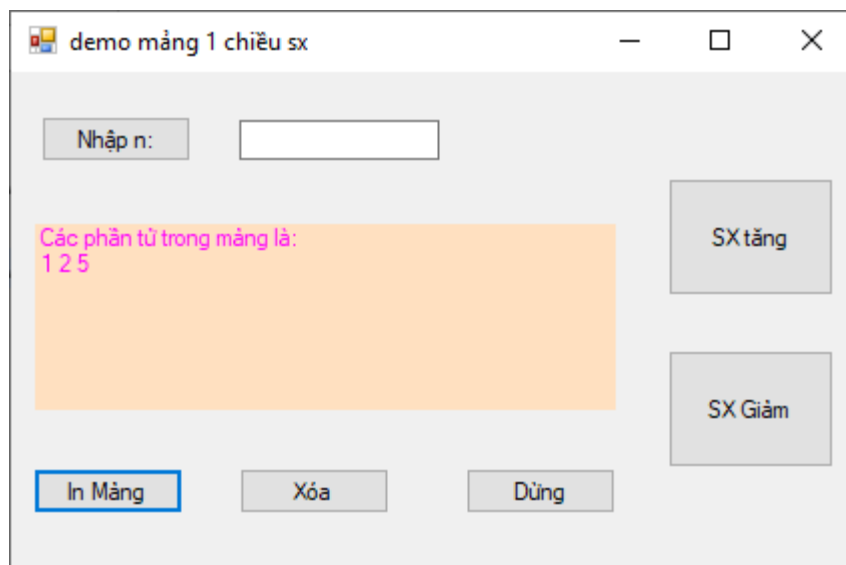
Nhấn nút in mảng



Nhấn nút SX tăng

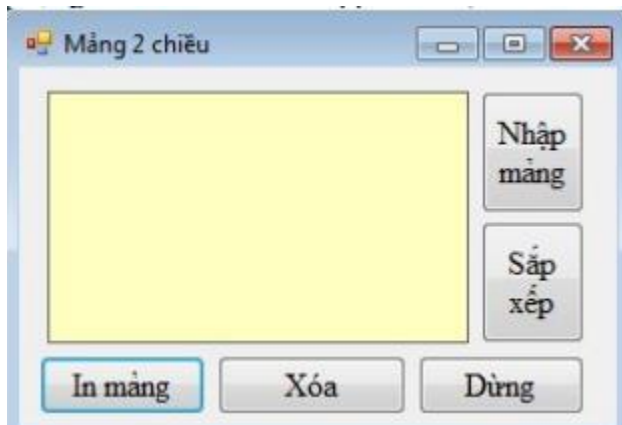


Nhấn nút in mảng



// làm tương tự cho nút giảm

Bài 3:



* Yêu cầu

- Thiết kế form như mẫu (lblKQ, btnIn, btnXoa, btnDung, btnNhap, btnSapXep).
- Khai báo mảng 2 chiều gồm 4 dòng, 3 cột chứa các số nguyên.
- Nhấp vào button **Nhập mảng** để nhập các phần tử cho mảng (có giá trị = dòng - cột).
- Nhấp vào button **Sắp Xếp** sẽ sắp xếp mảng tăng dần theo từng hàng.
- Nhấp vào button **In mảng** (btnIn) sẽ in ra label (lblKQ) các giá trị trong mảng.
- Nhấp vào button **Xóa** (btnXoa) sẽ xóa trống nội dung của label (lblKQ).
- Nhấp vào button **Dừng** sẽ dừng chương trình.

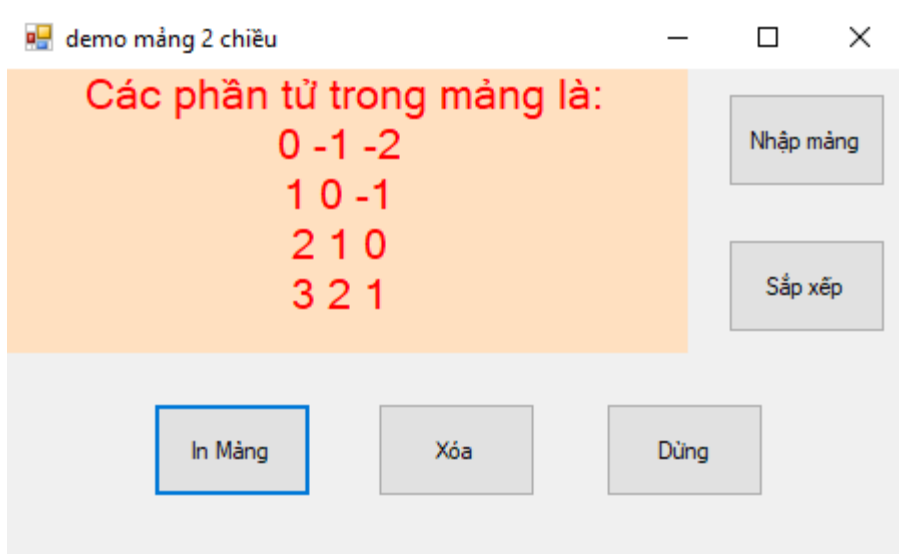
* Hướng dẫn

- Thiết kế form như yêu cầu.
- Khai báo mảng: qua code, thêm đoạn code để được kết quả như sau:

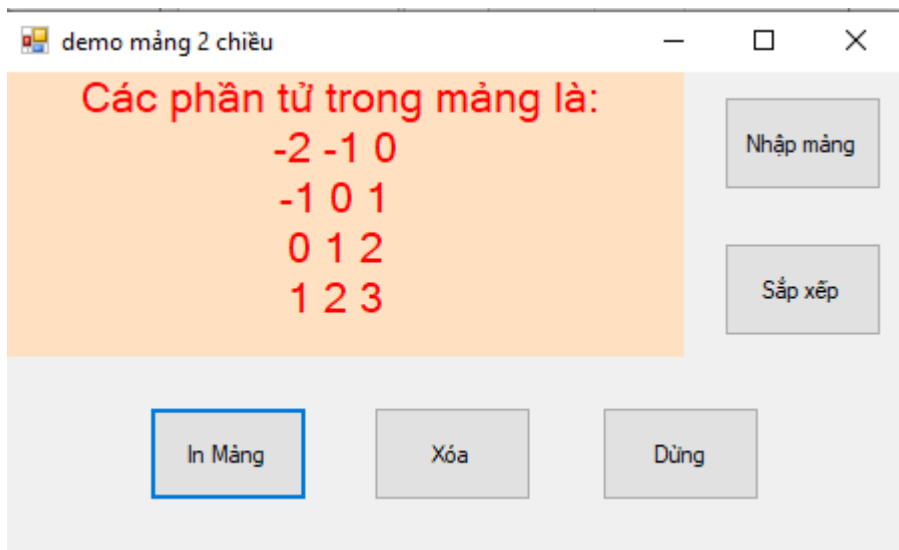
```
public partial class Form1 : Form
{
    // Khai báo 4 dòng 3 cột
    const int sodong = 4;
    const int socot = 3;
    // Khai báo mảng 2 chiều gồm 4 dòng, 3 cột chứa 12 số nguyên
    int[,] Array2 = new int[sodong, socot];
}
```

// sv tự thực hiện yêu cầu trên

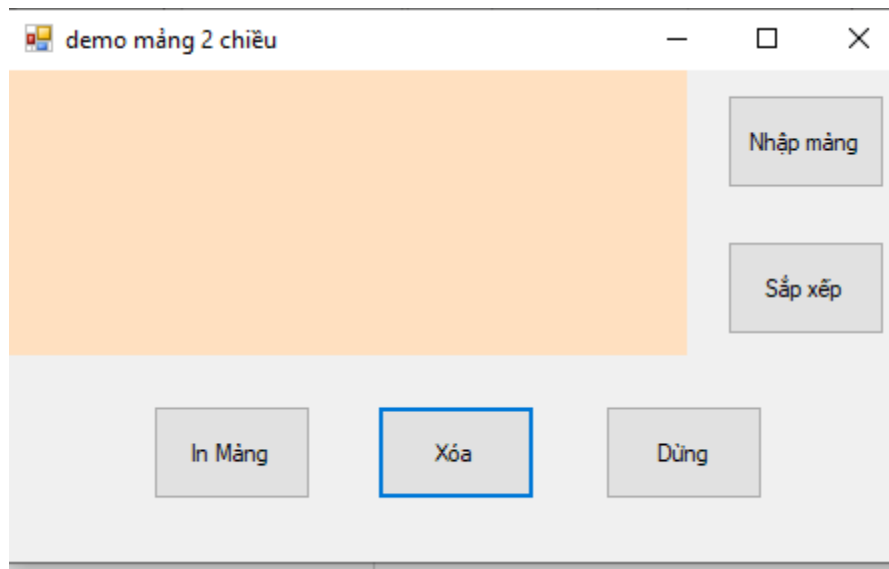
Nhấn nút nhập mảng, nhấn in mảng



Nhấn nút sắp xếp, in mảng



Nhấn nút xóa



Nhấn nút dừng

Bài 4:

Viết ứng dụng RandomNumber (chương trình game quay số: người chơi chọn một đáp án (3-10 hoặc 11-18)), sau đó nhấn nút quay số, các số sẽ phát sinh ngẫu nhiên. Nếu tổng ba số đúng trong khoản người chơi đã chọn, điểm được +10, ngược lại điểm bị -10. (điểm ban đầu bằng 0)

Random Number

2

0

3

Chọn

☐ 3-10

☒ 11-18

Quay số

10

Hướng dẫn:

Tạo form với giao diện sau:

- Sử dụng đối tượng Random để sinh số ngẫu nhiên.
- Double click vào nút quay số, viết code xử lý sự kiện click vào nút này.


```
int so1, so2, so3;
int tong;
Random rand = new Random();
//Random 3 số -> hiển thị ra form
so1 = rand.Next(0, 7); //0-6
so2 = rand.Next(7);
so3 = rand.Next(7);
lb1.Text = so1.ToString();
lb2.Text = so2.ToString();
lb3.Text = so3.ToString();
//So sánh tổng và sự lựa chọn người chơi
tong = so1 + so2 + so3;
if (rd310.Checked) //người chơi chọn rad310
{
    if (tong >= 3 && tong <= 10)
    {
        diem += 10;
    }
    else
    {
        diem -= 10;
    }
}
```

Bài 5:

Dental Payment Application

Dental Payment Form

Tên khách hàng

<input type="checkbox"/> Cạo vôi	\$100.000
<input type="checkbox"/> Tẩy trắng	\$1.200.000
<input type="checkbox"/> Chụp hình răng	\$200.000
Trám răng <input type="text" value="0"/>	\$80.000/cái

Total

Hướng dẫn:

Bảng 1: Mô tả các control trên form

STT	Name	Control	Thiết lập các thuộc tính cho control
1	lblTitle	Label	Text = “Dental Payment Form”, Font = “Impact, Size = 17”, ForeColor = Blue
2	lblName	Label	Text = “Tên khách hàng”
3	txtName	TextBox	
4	chkClean	CheckBox	Text = “Cạo vôi”
5	lblCleanCost	Label	Text=”\$100000”
6	chkWhitening	CheckBox	Text=”Tẩy trắng”
7	lblWhiteningCost	Label	Text=”\$1200000”
8	chkXRay	CheckBox	Text=”Chụp hình răng”
9	lblXRayCost	Label	Text=”\$200000”
10	lblFilling	Label	Text=”Trám răng”
11	numFilling	NumericUpDown	
12	lblFillCost	Label	Text=”\$80000”
13	lblTotal	Label	Text=”Total”
14	txtTotal	TextBox	Enable = False
15	btnExit	Button	Text=”Thoát”
16	btnCalc	Button	Text=”Tính tiền”

Bước 5. Tạo chức năng tính tiền, chức năng này được kích hoạt khi button “Tính tiền” được chọn.

Mô tả chức năng **GetPay()** như sau (GetPay()) là phương thức thành viên của lớp Form chính:

- + Kiểm tra xem tên khách hàng có được nhập hay không?
- + Nếu chưa: ➔ xuất thông báo, yêu cầu nhập tên khách.
- + Đã nhập: thực hiện các bước sau
 - Total = 0
 - If (cạo vôi) Total += 100.000
 - If (tẩy trắng) Total += 1.200.000
 - If (chụp hình răng) Total += 200000
 - Total += (số răng trám)*80000
 - Xuất số tiền ra TextBox txtTotal

Sinh viên tự viết phương thức GetPay theo mô tả bên trên!

Tạo trình xử lý sự kiện cho button btnCalc rồi trong trình xử lý sự kiện này gọi chức năng GetPay.

Kết quả chương trình:

Trường hợp không nhập tên khách hàng: phát sinh message box cảnh báo

Dental Payment Application

Dental Payment Form

Tên khách hàng


☐ Cạo vôi \$100.000

☐ Tẩy trắng

☐ Chụp hình răng

Trám răng

Thông báo

 Chưa nhập tên khách hàng!

OK

Thoát Tính tiền

Màn hình sau khi tính tiền cho khách

Dental Payment Application

Dental Payment Form

Tên khách hàng **Nguyen**

☒ Cạo vôi \$100.000

☐ Tẩy trắng \$1.200.000

☒ Chụp hình răng \$200.000

Trám răng \$80.000/cái

Total

Thoát Tính tiền

Bài tập làm thêm

Bài tập

Khởi tạo một ứng dụng Windows Forms Application:

1. Khai báo 1 mảng nguyên 1 chiều tối đa 10 phần tử. Viết chương trình:
 - Nhập vào giá trị cho 1 phần tử trong mảng.
 - In giá trị của các phần tử trong mảng.
 - In giá trị lớn nhất, giá trị nhỏ nhất của các phần tử trong mảng.
 - In tổng số các giá trị, trung bình cộng các giá trị của các phần tử trong mảng.
2. Khai báo 1 mảng nguyên 2 chiều 4 dòng, 5 cột. Viết chương trình:
 - Nhập giá trị cho các phần tử trong mảng (giá trị = số thứ tự dòng + số thứ tự cột).
 - In giá trị các phần tử trong mảng.
 - In giá trị lớn nhất, giá trị nhỏ nhất của các phần tử trong mảng.
 - In tổng số các giá trị, trung bình cộng các giá trị của các phần tử trong mảng.