



A. MỤC TIÊU:

- Thực hiện được Add control động.
- Tìm kiếm control trên Form.

B. DỤNG CỤ - THIẾT BỊ THỰC HÀNH CHO MỘT SV:

STT	Chủng loại – Quy cách vật tư	Số lượng	Đơn vị	Ghi chú
1	Computer	1	1	

C. NỘI DUNG THỰC HÀNH

1. Tóm tắt lý thuyết

Trong một chương trình xây dựng trên nền tảng C#, đôi khi cần phải sử dụng tới các controls động hay còn gọi là Dynamic Control, nghĩa là chúng ta không sử dụng (kéo thả) các controls từ thanh Toolbox ngay ở bước thiết kế giao diện, mà sẽ thiết kế chúng bằng code, khi chương trình chạy code sẽ thực thi và controls sẽ được thêm và hiển thị.

Lợi ích của Controls động trong C# là người dùng có thể tạo ra chúng khi tương tác với chương trình và có thể tạo ra bất cứ Control động nào có trong C#.

Phát sinh Control thêm tự động lên Form

1.1. Quá trình phát sinh Control tự động thêm lên Form

1. Khởi tạo control
2. Định kích thước cho control
3. Định vị trí cho control
4. Tùy chọn:
 - ✓ Đặt tên
 - ✓ Gắn sự kiện
 - ✓ ...
5. Add control vào Container (Form, Groupbox, Panel, ...)

1.2. Ví dụ

```
Button b = new Button();  
b.Name = "b1";  
b.Size = new Size(30, 30);  
b.Left = 30;  
b.Top = 20;  
this.Controls.Add(b);
```

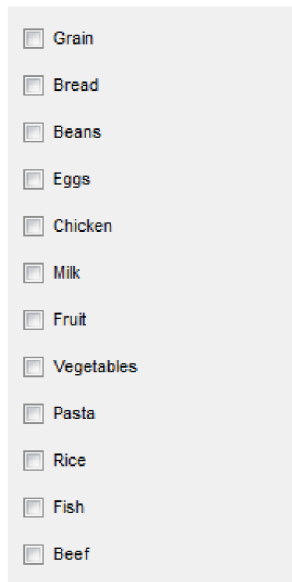
```
b.Click += new System.EventHandler(bt_Click);
private void bt_Click(object sender, EventArgs e)
{
}
}
```

1.3. Tìm kiếm Control trên Form. Ví dụ tìm kiếm Control là loại TextBox trên một đối tượng Container

```
foreach (Control item in ctr.Controls)
{
    if (ctr.GetType() == typeof(TextBox))
    {
        ctr.Text = "";
    }
}
```

2. Bài tập hướng dẫn mẫu

Bài 1. Tạo danh sách checkbox cho người dùng tùy chọn thêm danh sách thức ăn như hình sau:



Bước 1. Xác định thời điểm tạo Control để lựa chọn Event cho phù hợp

- Khi Form Load hoặc Khi kích chọn.

Bước 2. Xác định loại control và số lần thực hiện tạo loại Control đó, đồng thời xác định loại vòng lặp để thực hiện việc có khởi tạo hàng loạt Control hay không.

- Theo danh sách định trước, vòng lặp For hoặc Foreach

Bước 3. Thực hiện thêm Control trong sự kiện phù hợp và lặp đi lặp lại quá trình tạo hàng loạt Control đó

Lặp với số lần cần thiết quá trình sau:

1. Khởi tạo control
2. Định kích thước cho control
3. Định vị trí cho control

4. Tùy chọn:

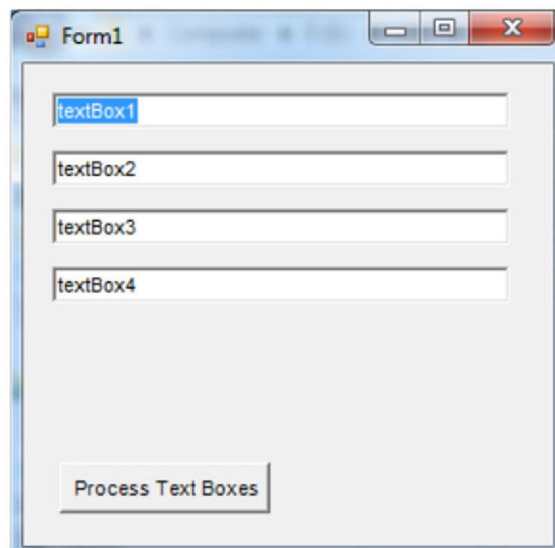
- ✓ Đặt tên
- ✓ Gắn sự kiện
- ✓

5. Add control vào Container (Form, Groupbox, Panel, ...)

Lập với số lần cần thiết quá trình sau:

```
string[] foods = { "Grain", "Bread", "Beans", "Eggs", "Chicken",  
"Milk", "Fruit", "Vegetables", "Pasta", "Rice", "Fish", "Beef" };  
int topPosition = 10;  
foreach (string food in foods)  
{  
    CheckBox checkBox = new CheckBox();  
    checkBox.Left = 10;  
    checkBox.Top = topPosition;  
    topPosition += 30;  
    checkBox.Text = food;  
    Controls.Add(checkBox);  
}
```

Bài 2. Duyệt tìm kiếm Control Textbox chứa trong một container (có thể là form) và xóa tất cả thông tin trong các Textbox đó.



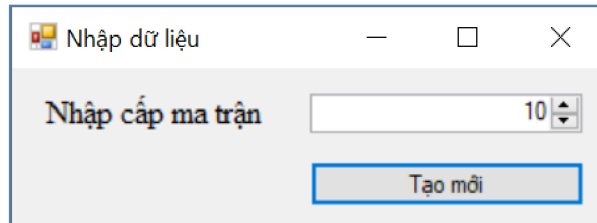
```
//Gợi ý:  
foreach (Control item in this.Controls) {  
    if (item.GetType() == typeof(TextBox))  
    {  
        item.Text = string.Empty;  
    }  
}
```

3. Bài tập tự làm

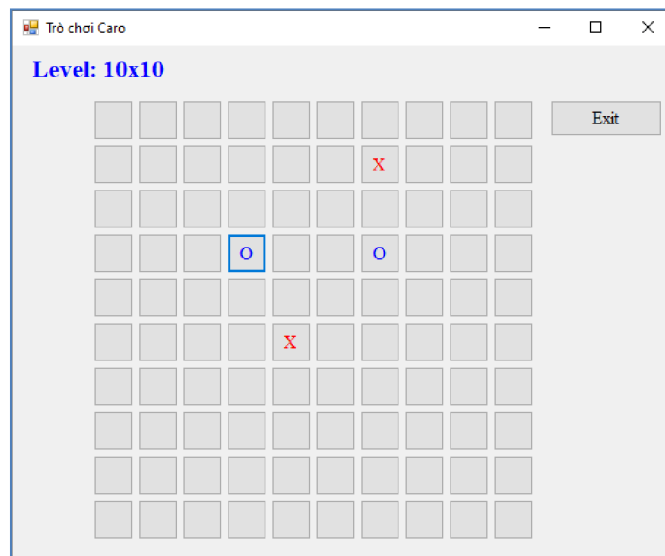
Bài 3. Tạo ứng dụng mô phỏng trò chơi Caro.

Yêu cầu:

1. Thiết kế màn hình trò chơi Caro
2. Cho phép người dùng nhập cấp ma trận vuông (cấp ma trận ≤ 50 .) của bàn cờ từ một form (dạng dialog) và chọn button Tạo mới.



Ví dụ: nhập vào cấp 10 thì ma trận sẽ là 10x10 (10 dòng 10 cột).



Bài 4. Xây dựng ứng dụng thi trắc nghiệm

Yêu cầu:

- File xml lưu trữ ngân hàng câu hỏi (ít nhất 5 câu hỏi). Dạng câu hỏi multiple choice và không giới hạn số lựa chọn (tối đa 6 lựa chọn) theo mẫu:

```
<?xml version="1.0" encoding="utf-8" ?>
<nganhang>
  <cauhoi ma="">
    <noidung></noidung>
    <dsluachon>
      <luachon dung="0"></luachon>
      <luachon dung="0"></luachon>
      <luachon dung="1"></luachon>
      <luachon dung="0"></luachon>
    </dsluachon>
  </cauhoi>
```

```

<cauhoi ma ="">
    <noidung></noidung>
    <dsluachon>
        <luachon dung="1"></luachon>
        <luachon dung="0"></luachon>
    </dsluachon>
</cauhoi>
</nganhang>

```

- Đọc câu hỏi trắc nghiệm từ ngân hàng câu hỏi, lưu trữ vào danh sách câu hỏi (List) và hiển thị lên form với giao diện mẫu:

- Tính điểm khi người dùng nộp bài hoặc hết giờ.

Bài 5. Xây dựng ứng dụng Quản lý phòng trọ

- Thêm thông tin phòng trọ mới theo tầng với thông tin: mã phòng, tình trạng phòng, sức chứa.
- Cập nhật tình trạng phòng trọ: trống, đã đặt, đã có người ở, đang sửa chữa bằng cách right click vào phòng đó và đổi màu tương ứng với tình trạng phòng.
- Click chọn phòng thì hiện thông tin của phòng đó lên MessageBox.

Bài 6. Xây dựng ứng dụng có giao diện như sau:

Yêu cầu:

1. Formload:
 - a. Listbox mặc định được chọn phần tử đầu tiên, danh sách được đọc từ file.

Gợi ý:

```
StreamReader f = new StreamReader("DM_TinhThanh.txt");
while (!f.EndOfStream)
{
    listBox1.Items.Add(f.ReadLine());
}
```

- b. Combobox cho phép AutoComplete
2. Tạo mới: Làm sạch các Textbox trên form
3. Lưu:
 - a. Họ tên là kí tự a, b, c...
 - b. Textbox số điện thoại chỉ cho phép nhập số

- c. Textbox Email bắt buộc phải có kí tự @
- d. Lưu dữ liệu xuống file text, mỗi thông tin trên một dòng

```
StreamWriter sw = StreamWriter("ThongTinSV.txt");  
sw.Write("string");  
sw.Close();
```

4. Bài tập về nhà

Bài 7. Hoàn chỉnh bài 3 với các yêu cầu sau:

- Vô hiệu hóa button khi người dùng đã click chọn
- Kiểm tra kết quả thắng/ thua và hiển thị kết quả
- Bổ sung thời gian chơi, giới hạn thời gian cho một lượt chơi là 30 giây (đồng hồ đếm ngược).

Bài 8. Tạo ứng dụng xây dựng màn hình Game trắc xanh. Ứng dụng cho phép nhập vào số cặp hình. Ví dụ nhập vào số 10 thì màn hình có 10 cặp hình. Mỗi dòng chứa 5 hình trên 4 dòng.

