

Hướng dẫn giải 1.3

- Trong thư mục Chuong1, anh/chị dùng Visual Studio 2017, tạo một đề án C#, cũng với loại Windows Form App (.NET framework). Lần này lấy tên đề án là Bai3.
- Sửa lại tên Form1 thành frmBai3. Sửa lại tiêu đề form này thành Bài toán 3.
- Thiết kế giao diện như sau:

STT	Tên đối tượng	Loại đối tượng	Text	Các thuộc tính khác	Vị trí
1	gbLoaiKhoa	GroupBox	Loại khóa		Ở dòng đầu tiên
2	gbPhuongPhapMa	GroupBox	Phuong pháp mã		Ngay dưới gbLoaiKhoa
3	lblBanTinRo	Label	Bản tin rõ		Ngay dưới gbPhuongPhapMa
4	txtBanTinRo	TextBox	(để trống)		Bên phải lblBanTinRo
5	lblBanTinDuocMaHoa	Label	Bản tin được mã hóa		Ngay dưới lblBanTinRo
6	txtBanTinDuocMaHoa	TextBox	(để trống)		Bên phải lblBanTinDuocMaHoa
7	lblBanTinDuocGiaiMa	Label	Bản tin được giải mã		Ngay dưới lblBanTinDuocMaHoa
8	txtBanTinDuocGiaiMa	TextBox	(để trống)		Bên phải lblBanTinDuocGiaiMa
9	btnMaHoa	Button	Mã hóa	Enabled=false	Ngay dưới lblBanTinDuocGiaiMa
10	btnGiaiMa	Button	Giải mã	Enabled=false	Bên phải btnMaHoa

- Bên trong gbLoaiKhoa có các đối tượng sau:

STT	Tên đối tượng	Loại đối tượng	Text	Các thuộc tính khác	Vị trí
1	rbSo	RadioButton	Số	Checked=true	Ở vị trí bên trái
2	rbChu	RadioButton	Chữ	Checked=false	Bên phải rbSo
3	txtMaKhoa	TextBox	(để trống)		Bên phải rbChu

5. Bên trong gbPhuongPhapMa có các đối tượng sau:

STT	Tên đối tượng	Loại đối tượng	Text	Các thuộc tính khác	Vị trí
1	rbDongVaCot	RadioButton	Dòng và cột	Checked=true	Ở vị trí bên trái
2	rbDuongCheo	RadioButton	Theo đường chéo	Checked=false	Bên phải rbDongVaCot

6. Trước hàm tạo của frmBai3, anh/chị khai báo các biến sau:

- Biến thứ nhất, có tên MaTran, với kiểu char[,] và được khởi tạo bằng hàm tạo ma trận mảng 8 dòng x 8 cột.
- Biến thứ hai, có tên KhoHopLe, với kiểu luận lý, và giá trị ban đầu là false.
- Biến thứ ba, có tên BanTinRoHopLe, với kiểu luận lý, và giá trị ban đầu là false.
- Biến thứ tư, có tên KhoaSo, với kiểu int[] và được khởi tạo bằng mảng 8 số nguyên, nhưng các phần tử mảng chưa được gán giá trị ban đầu.
- Biến thứ năm, có tên KhoaSoNguoc, với kiểu int[] và được khởi tạo bằng mảng 8 số nguyên, nhưng các phần tử mảng chưa được gán giá trị ban đầu.

7. Bên dưới hàm tạo frmBai3, anh/chị khai báo phương thức riêng tư, có tên KiemTraHopLe. Phương thức này có một tham biến hình thức, có tên s, với kiểu chuỗi. Phương thức này trả về kiểu Luận lý. Nhiệm vụ của phương thức này như sau:

```
bool HopLe = true;
for (int i = 0; i < s.Length; i++)
{
    char c = s[i];
    if (c < 'A' || c > 'Z')
    {
        HopLe = false;
        break;
    }
}
return HopLe;
```

8. Anh/chị khai báo một phương thức riêng tư, có tên KiemTraMaKhoaHopLe. Phương thức này không có tham biến hình thức. Phương thức này trả về kiểu luận lý. Nhiệm vụ của phương thức này là:

```
if (rbSo.Checked)
{
    bool[] CacSo = new bool[8];
    if (txtMaKhoa.Text.Length < 8) return false;
    for (int i = 0; i < txtMaKhoa.Text.Length; i++)
    {
        char c = txtMaKhoa.Text[i];
        if (c < '1' || c > '8')
            return false;
        else
            CacSo[(int)c - 49] = true;
    }
    for (int i = 0; i < 8; i++)
        if (!CacSo[i]) return false;
    for (int i = 0; i < 8; i++)
        KhoaSo[i] = txtMaKhoa.Text[i] - 49;
    return true;
}
else
{
    bool[] CacChu = new bool[26];
    if (txtMaKhoa.Text.Length < 8) return false;
    List<char> dsChu = new List<char>();
    for (int i = 0; i < txtMaKhoa.Text.Length; i++)
```

```

{
    char c = txtMaKhoa.Text[i];
    if (c < 'A' || c > 'Z')
        return false;
    else
    {
        if (dsChu.Contains(c)) return false;
        dsChu.Add(c);
    }
}
dsChu.Sort();
string ss = "";
for (int i = 0; i < txtMaKhoa.Text.Length; i++)
{
    KhoaSo[i] = dsChu.IndexOf(txtMaKhoa.Text[i]);
    ss += KhoaSo[i].ToString() + " ";
}
MessageBox.Show(ss);
return true;
}

```

9. Anh/chị khai báo một phương thức riêng tư, có tên TaoKhoaSoNguoc. Phương thức này không có tham biến hình thức. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```

for (int i = 0; i < 8; i++)
    KhoaSoNguoc[KhoaSo[i]] = i;

```

10. Anh/chị nhấp đôi vào txtBanTinRo, để có trình xử lý biến cố TextChanged, có tên txtBanTinRo_TextChanged. Phương thức này có hai tham biến hình thức. Tham biến hình thức thứ nhất có tên sender, với kiểu object. Tham biến hình thức thứ hai, có tên e, với kiểu EventArgs. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```

BanTinRoHopLe = KiemTraHopLe(txtBanTinRo.Text);
btnMaHoa.Enabled = KhoaHopLe && BanTinRoHopLe;

```

11. Anh/chị nhấp đôi vào txtBanTinDuocMaHoa, để có trình xử lý biến cố TextChanged, có tên txtBanTinDuocMaHoa_TextChanged. Phương thức này có hai tham biến hình thức. Tham biến hình thức thứ nhất có tên sender, với kiểu object. Tham biến hình thức thứ hai, có tên e, với kiểu EventArgs. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```

BanTinRoHopLe = KiemTraHopLe(txtBanTinDuocMaHoa.Text);
btnGiaiMa.Enabled = KhoaHopLe && BanTinRoHopLe;

```

12. Anh/chị nhấp đôi vào txtMaKhoa, để có trình xử lý biến cố TextChanged, có tên txtMaKhoa_TextChanged. Phương thức này có hai tham biến hình thức. Tham biến hình thức thứ nhất có tên sender, với kiểu object. Tham biến hình thức thứ hai, có tên e, với kiểu EventArgs. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```

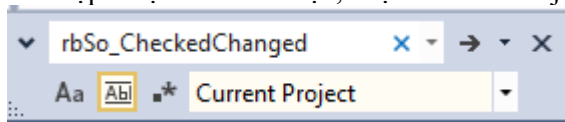
KhoaHopLe = KiemTraMaKhoaHopLe();
btnMaHoa.Enabled = KhoaHopLe && BanTinRoHopLe;

```

13. Anh/chị nhấp đôi vào rbSo, để có trình xử lý biến cố CheckChanged, có tên rbSo_CheckedChanged.

14. Anh/chị dùng con chuột chỉ vào dòng chữ rbSo_CheckChanged, bấm Ctrl-F.

15. Khi hộp thoại nhỏ xuất hiện, chọn Current Project như sau:



16. Bấm nút phải hai lần để đến dòng lệnh:

```

this.rbSo.CheckedChanged += new System.EventHandler(this.rbSo_CheckedChanged);

```

17. Anh/chị sửa lại dòng lệnh này như sau:

```

this.rbSo.CheckedChanged += new System.EventHandler(this.rb_CheckedChanged);

```

18. Anh/chị tìm phần cuối của các khai báo this.rbChu.

19. Anh/chị thêm vào một dòng lệnh:

```

this.rbChu.CheckedChanged += new System.EventHandler(this.rb_CheckedChanged);

```

20. Anh/chị quay lại phần code, sửa lại phương thức này như sau:

```
private void rb_CheckedChanged(object sender, EventArgs e)
{
}
}
```

21. Anh/chị thêm hi câu lệnh sau vào thân của phương thức này:

```
KhoaHopLe = KiemTraMaKhoaHopLe();
btnMaHoa.Enabled = KhoaHopLe && BanTinRoHopLe;
```

22. Anh/chị nhấp đôi vào btnMaHoa, để có trình xử lý biến cố Click, có tên btnMaHoa_Click. Phương thức này có hai tham biến hình thức. Tham biến hình thức thứ nhất có tên sender, với kiểu object. Tham biến hình thức thứ hai, có tên e, với kiểu EventArgs. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```
TaoKhoaSoNguoc();

for (int i = 0; i < 8; i++)
    for (int j = 0; j < 8; j++)
    {
        int ii = i * 8 + j;
        if (ii < txtBanTinRo.Text.Length)
            MaTran[i, j] = txtBanTinRo.Text[ii];
        else
            MaTran[i, j] = 'Z';
    }

// Ghép kết quả
string s = "";
if (rbDongVaCot.Checked)
{ // Theo dòng và cột
    for (int i = 0; i < 8; i++)
        for (int j = 0; j < 8; j++)
            s += MaTran[KhoaSo[i], KhoaSo[j]];
    txtBanTinDuocMaHoa.Text = s;
} else
{ // Theo đường chéo
    for (int ii=0; ii<15; ii++)
    {
        if (ii <= 7)
        {
            if (ii % 2 == 0)
            {
                for (int j = ii; j >= 0; j--)
                    s += MaTran[ii - j, KhoaSoNguoc[j]]; // Chéo xuống
            }
            else
            {
                for (int j = 0; j <= ii; j++)
                    s += MaTran[ii - j, KhoaSoNguoc[j]]; // Chéo lên
            }
        }
        else
        {
            if (ii % 2 == 0)
            {
                for (int j = 7; j >= ii - 7; j--)
                    s += MaTran[ii - j, KhoaSoNguoc[j]]; // Chéo xuống
            }
            else
            {

```

```

        for (int j=ii-7; j<=7; j++)
            s += MaTran[ii - j, KhoaSoNguoc[j]]; // Chéo lên
    }
}
// MessageBox.Show(s.Length.ToString());
}
txtBanTinDuocMaHoa.Text = s;
// MessageBox.Show(s.Length.ToString());
}

```

23. Anh/chị nhấp đôi vào btnGiaiMa, để có trình xử lý biến cố Click, có tên btnGiaiMa_Click. Phương thức này có hai tham biến hình thức. Tham biến hình thức thứ nhất có tên sender, với kiểu object. Tham biến hình thức thứ hai, có tên e, với kiểu EventArgs. Phương thức này không có kiểu dữ liệu trả về. Nhiệm vụ của phương thức này là:

```

TaoKhoaSoNguoc();
if (txtBanTinDuocMaHoa.Text.Length != 64)
{
    MessageBox.Show("Không thể giải mã được");
    return;
}

if (rbDongVaCot.Checked)
{ // Theo dòng và cột
    for (int i = 0; i < 8; i++)
        for (int j = 0; j < 8; j++)
        {
            int ij = i * 8 + j;
            MaTran[KhoaSo[i], KhoaSo[j]] = txtBanTinDuocMaHoa.Text[ij];
        }
} else
{ // Theo đường chéo
    int ij = 0;
    for (int ii = 0; ii < 15; ii++)
    {
        if (ii <= 7)
        {
            if (ii % 2 == 0)
            {
                for (int j = ii; j >= 0; j--)
                    MaTran[ii - j, KhoaSoNguoc[j]] = txtBanTinDuocMaHoa.Text[ij++]; // Chéo xuống
            }
            else
            {
                for (int j = 0; j <= ii; j++)
                    MaTran[ii - j, KhoaSoNguoc[j]] = txtBanTinDuocMaHoa.Text[ij++]; // Chéo lên
            }
        }
        else
        {
            if (ii % 2 == 0)
            {
                for (int j = 7; j >= ii - 7; j--)
                    MaTran[ii - j, KhoaSoNguoc[j]] = txtBanTinDuocMaHoa.Text[ij++]; // Chéo xuống
            }
            else
            {
                for (int j = ii - 7; j <= 7; j++)
                    MaTran[ii - j, KhoaSoNguoc[j]] = txtBanTinDuocMaHoa.Text[ij++]; // Chéo lên
            }
        }
    }
}

```

```

    }
    }
    // MessageBox.Show(s.Length.ToString());
}
}
string s = "";
for (int i = 0; i < 8; i++)
    for (int j = 0; j < 8; j++)
        s += MaTran[i, j];
txtBanTinDuocGiaiMa.Text = s;

```

24. Chạy thử chương trình, nản hình hiện ra:

The screenshot shows a Windows application window titled "Bài toán 3". It contains the following elements:

- Loại khóa (Key type):** Two radio buttons, "Số" (Number) which is selected, and "Chữ" (Letter). To the right is an empty text input field.
- Phương pháp mã (Encoding method):** Two radio buttons, "Dòng và cột" (Row and column) which is selected, and "Theo đường chéo" (Along the diagonal).
- Bản tin rõ (Plaintext):** An empty text input field.
- Bản tin được mã hóa (Encoded text):** An empty text input field.
- Bản tin được giải mã (Decoded text):** An empty text input field.
- Buttons:** Two buttons at the bottom, "Mã hóa" (Encode) and "Giải mã" (Decode), both of which are currently disabled (grayed out).

25. Chọn một hoán vị nào đó của 8 số từ 1 đến 8, chẳng hạn 47128635, mỗi số xuất hiện đúng một lần.

This screenshot shows the same application window as before, but with the following changes:

- The text input field next to the "Số" radio button now contains the permutation "47128635".
- The "Mã hóa" (Encode) and "Giải mã" (Decode) buttons remain disabled.

26. Nhập nội dung "HOATDONGBINHTHUONG" vào trong txtBanTinRo.

Bài toán 3

Loại khóa
☒ Số ☐ Chữ 47128635

Phương pháp mã
☒ Dòng và cột ☐ Theo đường chéo

Bản tin rõ HOATDONGBINHTHUONG

Bản tin được mã hóa

Bản tin được giải mã

Mã hóa Giải mã

27. Bấm nút btnMaHoa, có kết quả sau:

Bài toán 3

Loại khóa
☒ Số ☐ Chữ 47128635

Phương pháp mã
☒ Dòng và cột ☐ Theo đường chéo

Bản tin rõ HOATDONGBINHTHUONG

Bản tin được mã hóa ZZZZZZZZZZZZZZZTNHOGOADHUBIOHNTZZZZZZZZZZZZZZZZNGZZZZZZZZZZZZ

Bản tin được giải mã

Mã hóa Giải mã

28. Bấm nút btnGiaiMa, nội dung của txtBanTinDuocGiaiMa như sau:

Bài toán 3

Loại khóa
☒ Số ☐ Chữ 47128635

Phương pháp mã
☒ Dòng và cột ☐ Theo đường chéo

Bản tin rõ HOATDONGBINHTHUONG

Bản tin được mã hóa ZZZZZZZZZZZZZZZTNHOGOADHUBIOHNTZZZZZZZZZZZZZZZZNGZZZZZZZZZZZZ

Bản tin được giải mã HOATDONGBINHTHUONGZZ

Mã hóa Giải mã

(Phần cuối có dư ra các chữ Z cho đủ 64 ký tự)

29. Chọn nút bấm tròn rbDuongCheo, ta có kết quả

Bài toán 3

Loại khóa
☐ Số ☒ Chữ

Phương pháp mã
☐ Dòng và cột ☒ Theo đường chéo

Bản tin rõ

Bản tin được mã hóa

Bản tin được giải mã

33. Kết quả hiện ra thông điệp cho biết vị trí tương ứng với các số từ 0 đến 7 như sau:

0 3 2 4 7 6 1 5

34. Bấm nút OK, rồi bấm nút btnMaHoa, có kết quả như sau:

Bài toán 3

Loại khóa
☐ Số ☒ Chữ

Phương pháp mã
☐ Dòng và cột ☒ Theo đường chéo

Bản tin rõ

Bản tin được mã hóa

Bản tin được giải mã

35. Bấm nút btnGiaiMa để kiểm tra lại.

39. Anh/chị nhập “CHUNGTAVANHOCTOT” vào txtVanBanRo.

40. Anh/chị bấm btnMaHoa để có kết quả mã hóa.

41. Sau đó, anh/chị bấm btnGiaiMa để kiểm tra lại.

[illegible]

---oOo---