

# LAB 1. LÀM QUEN VỚI C# VÀ VISUAL STUDIO

THỜI LƯỢNG : 4 TIẾT

## A. Mục tiêu

- Giúp sinh viên làm quen với môi trường phát triển của Microsoft Visual Studio
- Ôn tập những kỹ thuật lập trình cơ bản và làm quen với ngôn ngữ C#
- Sau khi hoàn thành bài thực hành này, sinh viên cần:
  - Tạo được dự án dạng Console Application
  - Nắm vững các thao tác nhập-xuất dữ liệu
  - Hiểu rõ cách khai báo và sử dụng biến, hằng
  - Biết cách viết, tính toán và sử dụng các phép toán (toán tử)
  - Nắm vững cú pháp của các lệnh điều khiển và vận dụng để giải các bài toán
  - Nắm rõ và cài đặt được các thao tác trên mảng một chiều

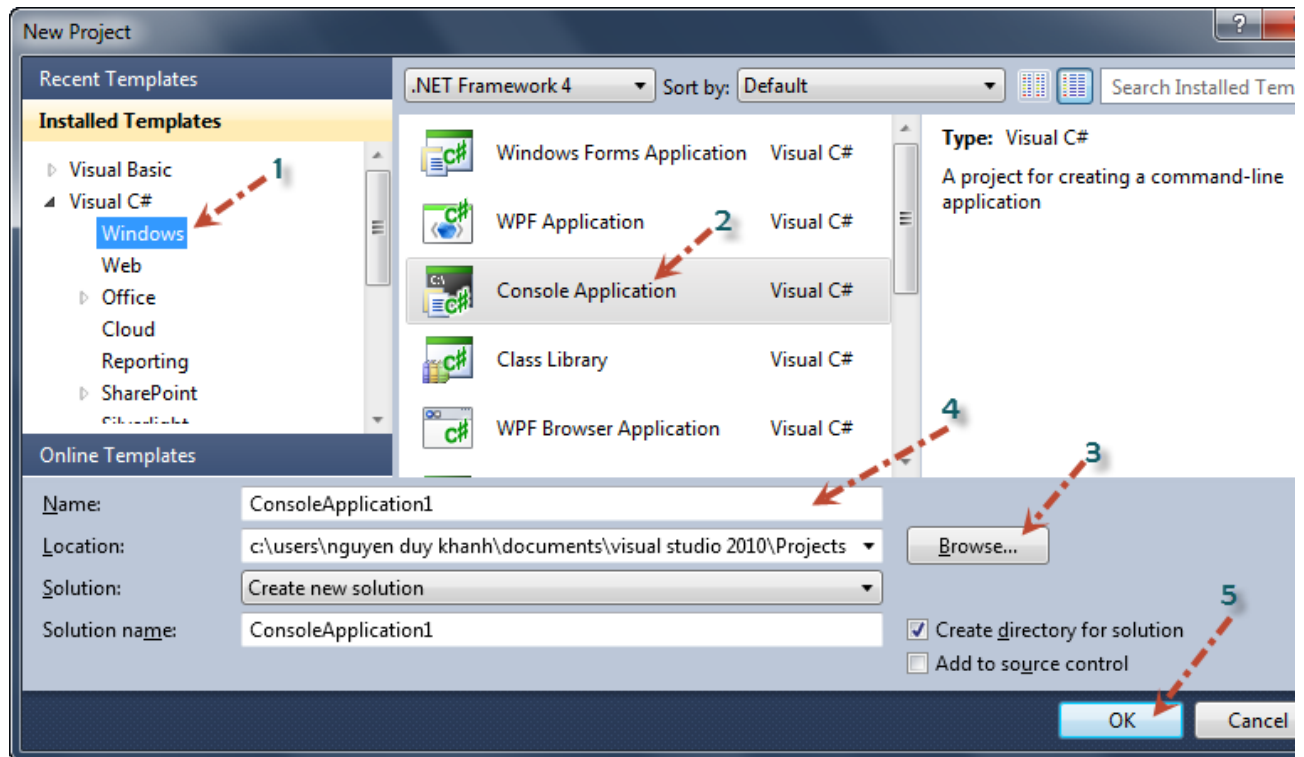
## B. Yêu cầu

- Sinh viên phải trả lời đầy đủ các câu hỏi (nếu có) hoặc chụp màn hình kết quả, ghi lại vào tập tin Word theo yêu cầu ghi trong phần hướng dẫn thực hành.
- Đặt tên tập tin Word theo dạng: Lab01\_MSSV\_HoVaTen.docx.
- Tạo thư mục, đặt tên là MSSV\_Lab01 để lưu tất cả bài làm trong bài thực hành này.
- Sinh viên phải hoàn thành tối thiểu 5 bài tập bắt buộc.
- Phải tạo một dự án riêng cho mỗi phần hoặc mỗi bài tập.
- Cuối buổi thực hành, nén thư mục trên & nộp bài qua email: [phucnguyen5555@gmail.com](mailto:phucnguyen5555@gmail.com)

## C. Hướng dẫn thực hành

### 1. Tạo dự án dạng Console Application

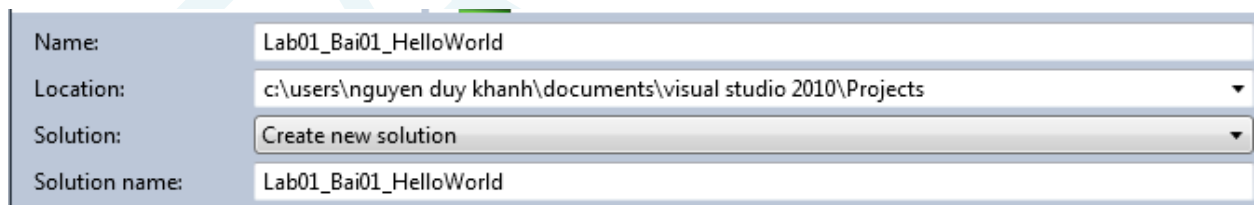
- Bước 1. Tạo một thư mục, đặt tên theo dạng MSSV-Lab01. Ví dụ: 1210234-Lab01.
- Bước 2. Mở chương trình Visual Studio 2010 bằng một trong những cách sau:
- Nhấp đôi chuột vào biểu tượng Microsoft Visual Studio trên Desktop
  - Start > All Programs > Microsoft Visual Studio ... > Microsoft Visual Studio...
  - Start > gõ Visual > nhấp chuột chọn Microsoft Visual Studio
- Bước 3. Vào menu File > New Project hoặc nhấn tổ hợp phím Ctrl+Shift+N để mở hộp thoại tạo dự án mới như sau:



Bước 4. Chọn các mục được đánh số 1 & 2 như trên hình.

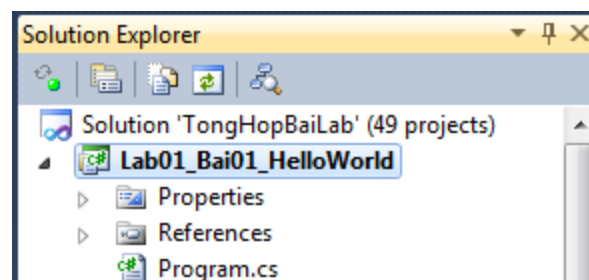
Bước 5. Nhấp chuột vào nút Browse để chọn đường dẫn tới thư mục đã tạo ở bước 1.

Bước 6. Trong khung Name, nhập tên của dự án. Lưu ý viết không dấu, không có khoảng trắng, bắt đầu bằng các chữ cái, chữ đầu tiên của mỗi từ viết hoa. Giả sử, tên dự án được đặt là Lab01\_Bai01\_HelloWorld như sau:

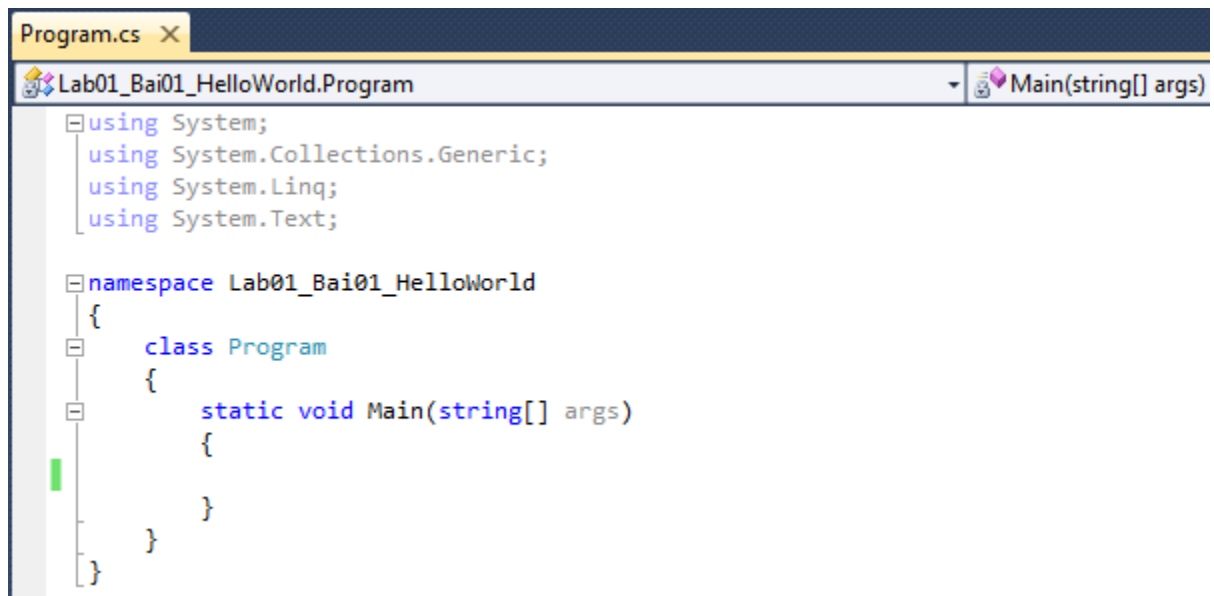


Bước 7. Nhấn nút OK để kết thúc việc tạo dự án và đóng hộp thoại.

Bước 8. Vào menu View > Solution Explorer hoặc nhấn tổ hợp phím Ctrl+Alt+L để mở cửa sổ hiển thị cấu trúc dự án như hình sau:



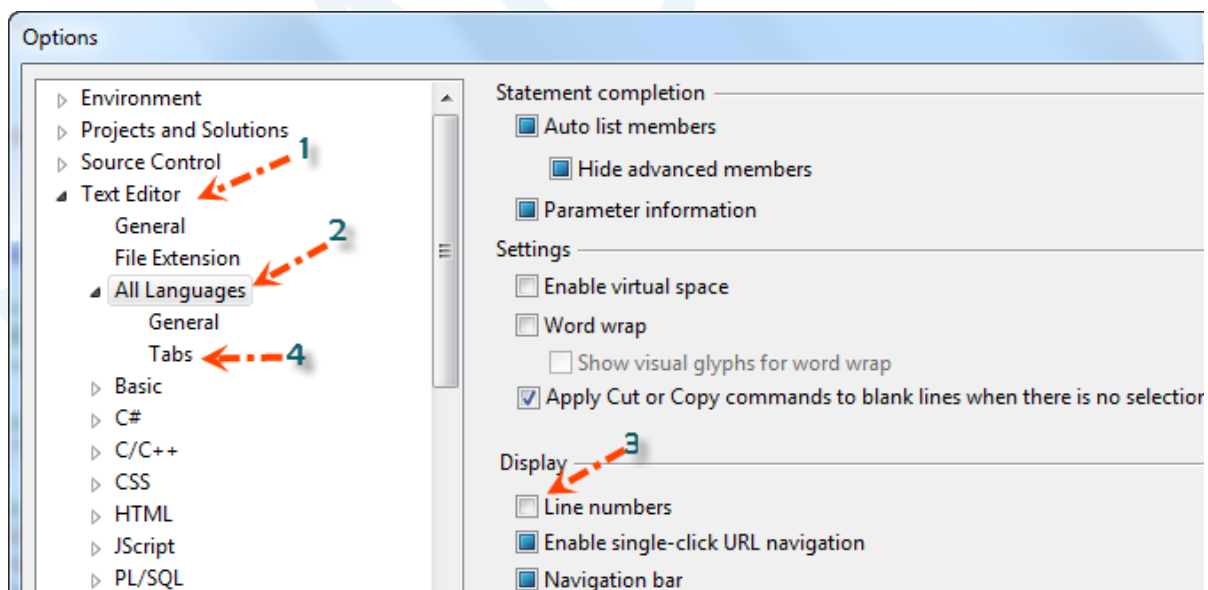
Bước 9. Nhấp đôi chuột vào tập tin Program.cs để mở cửa sổ soạn thảo.



Bước 10. Đến đây, dự án đã được tạo xong. Ta có thể viết mã lệnh vào giữa hàm Main.

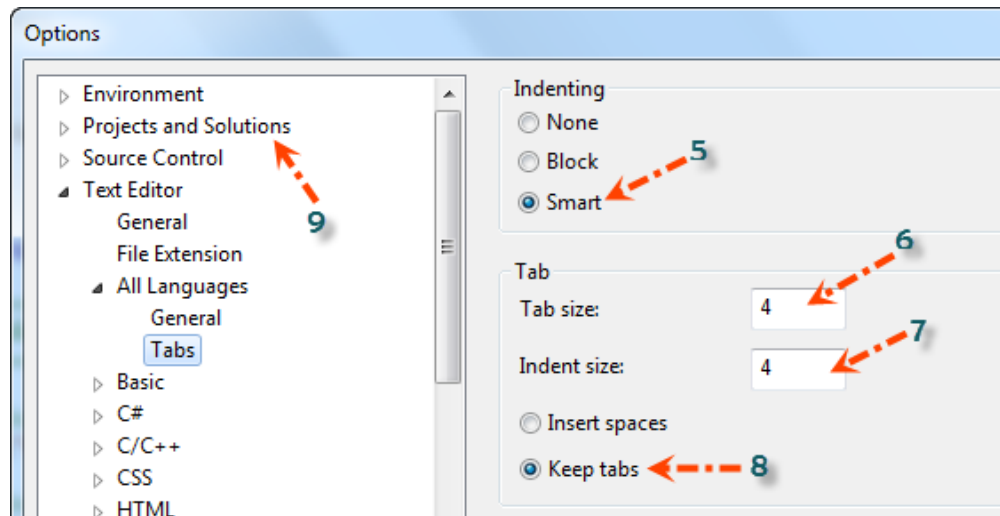
**Để tiện cho viết mã lệnh, ta thực hiện một số bước cấu hình như sau: (Chỉ làm một lần này, không cần phải làm lại cho các dự án khác)**

Bước 11. Vào menu Tools > Options > Chọn Text Editor > All Languages.

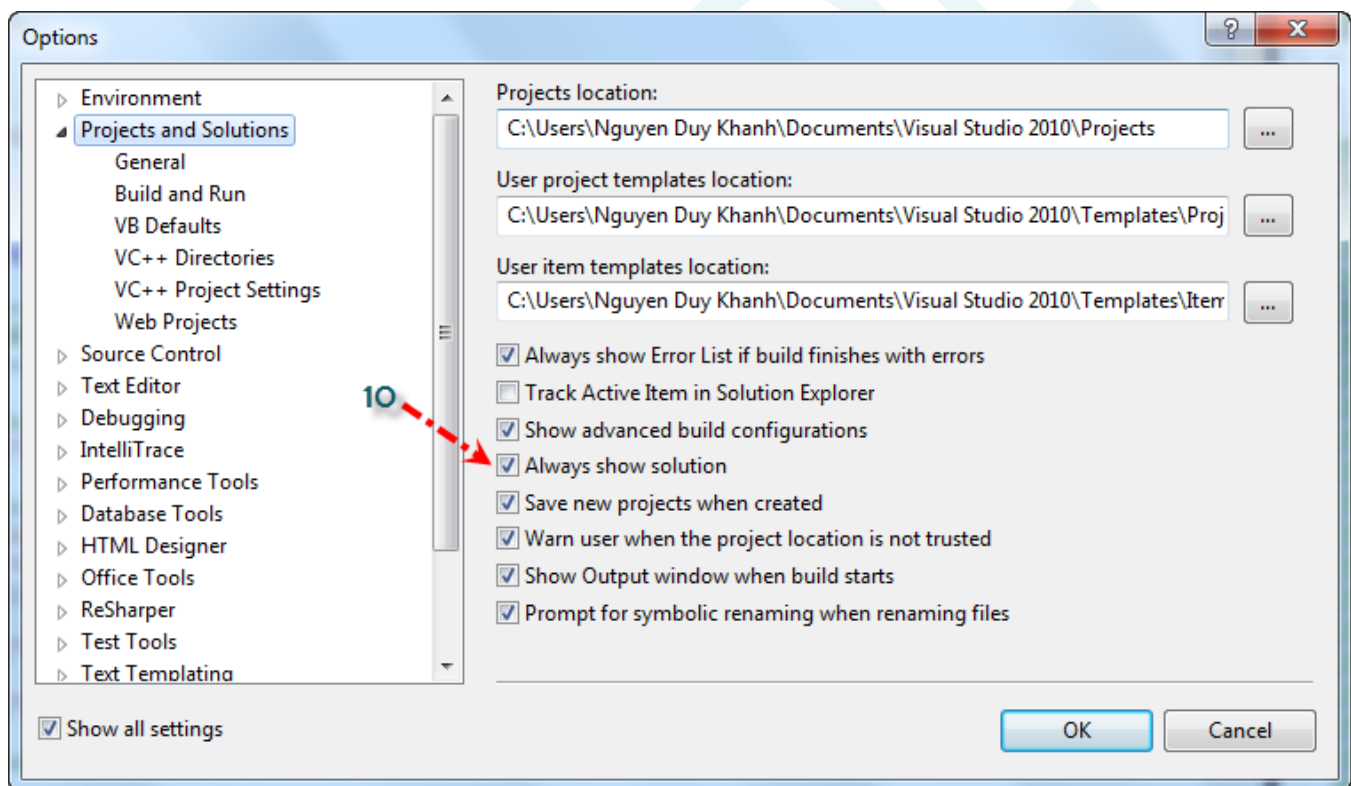


Bước 12. Đánh dấu chọn vào mục Line numbers. Sau đó chọn Tabs.

Bước 13. Trong mục Tabs, chọn Smart. Nhập Tab size = 4, Indent size = 4. Đánh dấu chọn Keep tabs. Sau đó chọn mục Projects and Solutions.



Bước 14. Trong mục Projects and Solutions, đánh dấu chọn Always show solution



Bước 15. Nhấn nút OK để kết thúc việc cấu hình. Trình soạn thảo sẽ hiển thị thêm số tự động ở cạnh bên trái. Khung Solution Explorer sẽ hiển thị tên của Solution.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5
6 namespace Lab01_Bai01_HelloWorld
7 {
8     class Program
9     {
10         static void Main(string[] args)
11         {
12
13         }
14     }
15 }

```

## 2. Viết chương trình Hello, World!

Trong phần này, ta sẽ viết một đoạn mã để xuất chuỗi "Hello, World!" ra màn hình theo nhiều cách.

Bước 1. Trong trình soạn thảo, tại dòng 12 như hình trên, nhập đoạn mã sau:

```
// Xuất chuỗi Hello, World! ra màn hình sau đó xuống dòng
Console.WriteLine("Hello, World!");

// Dừng chương trình và chờ nhấn 1 phím để kết thúc
Console.ReadKey();
```

Bước 2. Nhấn tổ hợp phím Ctrl+Shift+B hoặc vào menu Build để biên dịch chương trình.

Bước 3. Quan sát thông báo hiển thị ở góc dưới bên trái cửa sổ Visual Studio để biết kết quả. Nếu hiển thị thông báo Build succeeded nghĩa là không có lỗi. Ngược lại sẽ có dòng thông báo Build failed.

Bước 4. Trong trường hợp bị lỗi, vào menu View > Error List để mở cửa sổ hiển thị danh sách lỗi. Nhấp đôi chuột vào từng thông báo lỗi trong Error List để đến vị trí bị lỗi và sửa lỗi.

Bước 5. Trường hợp biên dịch thành công, nhấn phím F5 hoặc vào menu Debug > Start debugging hoặc nhấn nút Start debugging (F5) trên thanh công cụ để chạy chương trình.

Bước 6. Màn hình kết quả (màu đen) xuất hiện, hãy ghi nhận lại kết quả.

Bước 7. Hãy thay đổi đoạn mã trong bước 1 bởi đoạn mã sau và chạy lại chương trình:

```
// Xuất chuỗi Hello, sau đó xuất tiếp chuỗi World!
Console.Write("Hello, ");
Console.Write("World!");
Console.WriteLine(); // xuất dòng trống và xuống dòng
```

Bước 8. Kết quả có khác gì với bước 6 hay không? Ghi lại kết quả.

Bước 9. Tiếp tục thay đổi đoạn mã trong bước 7 bởi đoạn mã sau và chạy lại chương trình:

```
Console.WriteLine("Hello, " + "World!");  
  
Console.WriteLine("Da xuat chuoai Hello, World! " + 3 + " lan roi");
```

Bước 10. Kết quả có khác gì với bước 6 & 8 hay không? Ghi lại kết quả. Giải thích 2 lệnh đó.

### 3. Khai báo biến và hằng

Phần này hướng dẫn cách khai báo các hằng số và các biến với nhiều kiểu dữ liệu khác nhau. **Chú ý, những đoạn chữ bắt đầu với // có màu xanh là chú thích, không cần gõ lại những đoạn chữ này.**

Bước 1. Tạo một dự án mới, đặt tên là Lab01\_Bai02\_KhaiBaoBienVaHang

Bước 2. Mở tập tin Program.cs và nhập đoạn mã sau vào giữa hàm Main

```
// Định nghĩa hằng số  
// Quy tắc chung: const Kiểu_Du_Lieu TEN_HANG = Gia_Tri;  
  
const int THANG = 12;           // Số tháng trong năm  
const float TOC_DO_CPU = 2.4f; // Tốc độ CPU  
const double PI = 3.14159;      // Số PI  
const char TAB = '\t';          // Phím TAB  
const string KHOA = "Cong nghe Thong tin";  
  
// Xuất giá trị hằng ra màn hình  
Console.WriteLine("So thang trong nam : {0}", THANG);  
Console.WriteLine("Toc do CPU : {0} Ghz", TOC_DO_CPU);  
Console.WriteLine("Gia tri cua so PI : {0}", PI);  
Console.WriteLine("Ky tu (phím) TAB : {0}", TAB);  
Console.WriteLine("Khoa toi dang hoc : {0}", KHOA);  
  
// Dừng chương trình và chờ nhấn 1 phím để kết thúc  
Console.ReadKey();
```

Bước 3. Chạy chương trình và chụp hình, ghi nhận lại kết quả.

Bước 4. Trước dòng lệnh Console.ReadKey(); bổ sung đoạn mã sau:

```
// Khai báo biến  
// Quy tắc chung : Kiểu_Du_Lieu ten_Bien;  
// Khởi tạo giá trị: ten_Bien = Gia_Tri;  
  
string hoVaTen;           // Khai báo 1 biến kiểu chuỗi  
hoVaTen = "Trình Hoa Binh"; // Khởi tạo giá trị cho biến
```

```

int tuoi; // Khai báo biến kiểu số nguyên
tuoi = 20; // Khởi gán giá trị cho biến

// Khai báo và khởi tạo biến cùng lúc
// Quy tắc chung : Kieu_Du_Lieu ten_Bien = Gia_Tri;

float diemTrungBinh = 7.5f;
char gioiTinh = 'M'; // M = Male = Nam, F = Female = Nữ
bool daTotNghiep = false;
double chieuCao = 1.65; // Biến chiều cao, tính theo met

// Xuất giá trị các biến ra màn hình
Console.WriteLine("Ho va ten : {0}", hoVaTen);
Console.WriteLine("Tuoi : {0}", tuoi);
Console.WriteLine("Gioi tinh : {0}", gioiTinh);
Console.WriteLine("Chieu cao : {0} met", chieuCao);
Console.WriteLine("Diem T.Binh : {0}", diemTrungBinh);
Console.WriteLine("Da tot nghiep: {0}", daTotNghiep);

```

Bước 5. Chạy chương trình và chụp hình, ghi nhận lại kết quả.

Bước 6. Nhập tiếp đoạn mã sau:

```

// Có thể khai báo đồng thời nhiều biến cùng kiểu dữ liệu
double x, y; // Khai báo nhưng không khởi tạo
int p = 2, q = 3; // Vừa khai báo, vừa khởi tạo
char c, k = 'm'; // Cả 2

```

Bước 7. Viết lệnh xuất giá trị các biến & cho biết chương trình thực thi được không. Tại sao?

Bước 8. Hãy sửa lỗi (nếu có) và sau đó chạy chương trình để ghi nhận lại kết quả.

#### 4. Nhập xuất đơn giản

Phần này hướng dẫn thực hiện các thao tác nhập giá trị từ bàn phím, lưu giá trị nhập vào biến và sau đó xuất giá trị của biến ra màn hình.

Bước 1. Tạo một dự án mới, đặt tên là Lab01\_Bai03\_NhapXuat

Bước 2. Nhập đoạn mã sau vào giữa hàm Main trong tập tin Program.cs

```

Console.Write("Nhap ho va ten cua ban : ");

string hoTen; // Khai báo 1 biến kiểu chuỗi
hoTen = Console.ReadLine(); // Nhập giá trị từ bàn phím

// xuất chuỗi đã nhập ra màn hình
Console.WriteLine("Chuoi ban da nhap la : {0}", hoTen);

Console.ReadKey();

```

Bước 3. Chạy chương trình và chụp hình, ghi nhận lại kết quả. Mô tả lại hoạt động của chương trình.

Bước 4. Trước dòng lệnh `Console.ReadKey()`; bổ sung đoạn mã sau:

```
// Nhập một ký tự từ bàn phím
Console.Write("Nhap gioi tinh cua ban {Nam = M, Nu = F}:");
char gioiTinh; // Khai báo 1 biến ký tự

// Lấy 1 ký tự nhập vào từ bàn phím
gioiTinh = Console.ReadKey().KeyChar;

// Xuất ký tự vừa nhập ra màn hình
Console.WriteLine("Gioi tinh cua ban la : {0}", gioiTinh);
```

Bước 5. Chạy chương trình, chụp hình, ghi nhận lại kết quả & hoạt động của chương trình.

Bước 6. Tiếp tục bổ sung đoạn mã sau và cho biết kết quả của chương trình.

```
// Nhập một số nguyên từ bàn phím
Console.Write("Nhap nam sinh cua ban : ");
int namSinh; // Khai báo 1 biến kiểu số nguyên

// Nhập giá trị từ bàn phím, chuyển sang số nguyên
namSinh = int.Parse(Console.ReadLine());

// Xuất năm sinh đã nhập ra màn hình
Console.WriteLine("Ban sinh nam : {0}", namSinh);
```

Bước 7. Bằng cách tương tự, hãy viết mã để nhập vào một giá trị số thực cho biết điểm trung bình của sinh viên. Biết lệnh để chuyển sang số thực là `double.Parse` hoặc `float.Parse`.

## 5. Các phép toán và biểu thức

Phần này hướng dẫn cách sử dụng các toán tử (phép toán) số học, toán tử so sánh, toán tử thao tác bit, toán tử gán và các toán tử logic.

Bước 1. Tạo một dự án mới, đặt tên là `Lab01_Bai04_CacPhepToan`

Bước 2. Trong hàm `Main`, nhập đoạn mã sau:

```
int x = 12, y = 5, ketQua;

// =====
// Các phép toán số học
ketQua = x + y; // Phép cộng
Console.WriteLine("{0} + {1} = {2}", x, y, ketQua);

ketQua = x - y; // Phép trừ
Console.WriteLine("{0} - {1} = {2}", x, y, ketQua);
```



```

ketQua = x * y;           // Phép nhân
Console.WriteLine("{0} * {1} = {2}", x, y, ketQua);

ketQua = x / y;           // Phép chia
Console.WriteLine("{0} / {1} = {2}", x, y, ketQua);

ketQua = x % y;           // Phép chia lấy dư
Console.WriteLine("{0} % {1} = {2}", x, y, ketQua);

```

Bước 3. Chạy chương trình và ghi nhận, giải thích kết quả.

Bước 4. Thay đổi đoạn mã ở bước 2 thành đoạn mã sau

```

// =====
// Phép toán tự tăng/tự giảm
ketQua = x++;             // Tăng sau khi sử dụng
Console.WriteLine("x = {0}, Ket qua = {1}", x, ketQua);

ketQua = ++y;             // Tăng trước khi sử dụng
Console.WriteLine("y = {0}, Ket qua = {1}", y, ketQua);

ketQua = x--;             // Giảm sau khi sử dụng
Console.WriteLine("x = {0}, Ket qua = {1}", x, ketQua);

ketQua = --y;             // Giảm trước khi sử dụng
Console.WriteLine("y = {0}, Ket qua = {1}", y, ketQua);

```

Bước 5. Chạy chương trình và ghi nhận, giải thích kết quả.

Bước 6. Thay đổi đoạn mã ở bước 4 thành đoạn mã minh họa các toán tử gán

```

// =====
// Các toán tử thao tác bit
ketQua = ~x;              // Phép đảo bit 0-1
Console.WriteLine("y = {0}, Ket qua = {1}", y, ketQua);

ketQua = x | y;           // Phép OR theo bit
Console.WriteLine("{0} | {1} = {2}", x, y, ketQua);

ketQua = x & y;           // Phép AND theo bit
Console.WriteLine("{0} & {1} = {2}", x, y, ketQua);

ketQua = x << y;          // Phép dịch trái
Console.WriteLine("{0} << {1} = {2}", x, y, ketQua);

ketQua = x >> y;          // Phép dịch phải
Console.WriteLine("{0} >> {1} = {2}", x, y, ketQua);

```

Bước 7. Chạy chương trình và ghi nhận, giải thích kết quả.

Bước 8. Thay đổi đoạn mã ở bước 6 thành đoạn mã minh họa các toán tử so sánh

```
// =====
// Các toán tử so sánh
bool logic = false;

logic = x < y;           // Nhỏ hơn
Console.WriteLine("{0} < {1} : {2}", x, y, logic);

logic = x <= y;          // Nhỏ hơn hoặc bằng
Console.WriteLine("{0} <= {1} : {2}", x, y, logic);

logic = x > y;           // Lớn hơn
Console.WriteLine("{0} > {1} : {2}", x, y, logic);

logic = x >= y;          // Lớn hơn hoặc bằng
Console.WriteLine("{0} >= {1} : {2}", x, y, logic);

logic = x == y;          // So sánh bằng
Console.WriteLine("{0} == {1} : {2}", x, y, logic);

logic = x != y;          // Khác nhau
Console.WriteLine("{0} != {1} : {2}", x, y, logic);
```

Bước 9. Chạy chương trình và ghi nhận, giải thích kết quả.

Bước 10. Thay đổi đoạn mã ở bước 8 thành đoạn mã minh họa các toán tử gán sau

```
// =====
// Các toán tử gán

Console.WriteLine("x = {0}, y = {1}", x, y);

x += y;           // x = x + y
Console.WriteLine("x = {0}, y = {1}", x, y);

x -= y;           // x = x - y
Console.WriteLine("x = {0}, y = {1}", x, y);

x *= y;           // x = x * y
Console.WriteLine("x = {0}, y = {1}", x, y);

x /= y;           // x = x / y
Console.WriteLine("x = {0}, y = {1}", x, y);

x %= y;           // x = x % y
Console.WriteLine("x = {0}, y = {1}", x, y);

x <<= y;          // x = x << y
Console.WriteLine("x = {0}, y = {1}", x, y);
```

Bước 11. Chạy chương trình và ghi nhận, giải thích kết quả.

Bước 12. Thay đổi mã chương trình như sau và ghi nhận lại kết quả, giải thích kết quả đó.

```

bool p = true, q = false;

logic = !p;           // Toán tử phủ định
Console.WriteLine("!{0} = {1}", p, logic);

logic = p && q;
Console.WriteLine("{0} && {1} = {2}", p, q, logic);

logic = p || q;
Console.WriteLine("{0} || {1} = {2}", p, q, logic);

```

## 6. Các câu lệnh điều khiển

Phần này minh họa các cấu trúc điều khiển như lệnh IF, FOR, WHILE, DO...WHILE, SWITCH

Bước 1. Tạo một dự án mới, đặt tên là Lab01\_Bai05\_CauTrucDieuKien

Bước 2. Nhập đoạn mã sau vào hàm Main

```

// Khai báo biến "giờ hiện tại" và nhập giá trị từ bàn phím
Console.WriteLine("Nhập giờ hiện tại (0..24): ");

int gioHienTai;
gioHienTai = int.Parse(Console.ReadLine());

// =====
// Lệnh IF
if (gioHienTai % 2 == 0)
    Console.WriteLine("Gio chan");
else
    Console.WriteLine("Gio le");

if (gioHienTai < 0 || gioHienTai > 24)
    Console.WriteLine("Ban nhap gio sai");
else if (gioHienTai <= 12)
    Console.WriteLine("Chao buoi sang");
else
    Console.WriteLine("Chao buoi chieu");

```

Bước 3. Chạy chương trình, cho biết kết quả và mô tả lại hoạt động của chương trình.

Bước 4. Bổ sung đoạn mã sau vào cuối hàm Main. Đoạn mã này nhằm tính tổng n số nguyên dương đầu tiên.

```

// Lệnh FOR
int n = 10, tong = 0;
// Tính tổng các số từ 1 đến n
for (int i = 1; i <= n; i++)
    tong += i;

// Xuất kết quả ra màn hình
Console.WriteLine("1 + 2 + ... + {0} = {1}", n, tong);

```

Bước 5. Chạy chương trình và ghi nhận lại kết quả

Bước 6. Tiếp theo, ta sẽ giải bài toán sau: Anh BTX vay ngân hàng một số tiền là 25000000 và trả góp mỗi tháng 2000000, lãi suất 1%. Hỏi sau bao lâu anh BTX sẽ trả hết nợ.

```
double soTienVay = 25000000, laiSuat = 0.01, tienTra = 2000000;
int soThang = 0;

while (soTienVay > 0)
{
    soThang++;
    soTienVay *= 1 + laiSuat;
    soTienVay -= tienTra;
}

Console.WriteLine("So thang phai tra tien : {0}", soThang);
```

Bước 7. Chạy chương trình và xem kết quả.

Bước 8. Thay thế đoạn mã trong bước 6 bởi đoạn mã sau để bắt buộc nhập một số từ 0 -> 10:

```
int number = 0;

do
{
    Console.Write("Nhap mot so tu 0 den 10 : ");
    number = int.Parse(Console.ReadLine());
} while (number < 0 || number > 10);
```

Bước 9. Chạy chương trình và mô tả lại hoạt động của nó.

Bước 10. Thay thế đoạn mã ở bước 8, bởi đoạn mã sau:

```
char diem = 'B';
switch (diem)
{
    case 'A':
        Console.WriteLine("Hoc luc GIOI");
        break;
    case 'B':
        Console.WriteLine("Hoc luc KHA");
        break;
    case 'C':
        Console.WriteLine("Hoc luc TRUNG BINH");
        break;
    case 'D':
        Console.WriteLine("Hoc luc YEU");
        break;
    default:
        Console.WriteLine("Khong xep loai");
        break;
}
```

Bước 11. Chạy chương trình và cho biết kết quả.

## 7. Các thao tác trên mảng một chiều

Phần này hướng dẫn một số thao tác phổ biến trên mảng một chiều chứa các số nguyên.

Bước 1. Tạo một dự án mới, đặt tên là Lab01\_Bai06\_MangMotChieu

Bước 2. Nhập đoạn mã sau vào hàm Main

```
// Khai báo mảng một chiều tên là mmc chứa các số nguyên
int[] mmc;

// Khởi tạo mảng trống có thể chứa 1000 phần tử
mmc = new int[1000];

// Khai báo biến lưu số lượng phần tử đang có trong mảng
int dem = 0;
```

Bước 3. Tiếp tục nhập đoạn mã sau để nhập số lượng phần tử và nhập giá trị các phần tử.

```
// Yêu cầu người dùng nhập số phần tử
Console.WriteLine("Hay nhap so phan tu cua mang : ");
dem = int.Parse(Console.ReadLine());

// Nhập các phần tử của mảng
for (int i = 0; i < dem; i++)
{
    Console.WriteLine("a[{0}] = ", i);
    mmc[i] = int.Parse(Console.ReadLine());
}
```

Bước 4. Chạy chương trình, cho biết kết quả và nêu hoạt động của chương trình.

Bước 5. Bổ sung đoạn mã sau để xuất các phần tử của mảng ra màn hình

```
// Xuất các phần tử của mảng ra màn hình
Console.WriteLine("Cac phan tu cua mang : ");
for (int i = 0; i < dem; i++)
{
    Console.WriteLine("{0}\t", mmc[i]);
}
```

Bước 6. Chạy chương trình, chụp hình và ghi nhận lại kết quả

Bước 7. Trong đoạn mã ở bước 5, nếu thay hàm Write bởi WriteLine thì kết quả có gì khác?

Bước 8. Tiếp theo, ta sẽ viết mã lệnh để tìm phần tử có giá trị nhỏ nhất trong mảng. Viết tiếp đoạn mã sau vào hàm Main:

```
// Tìm phần tử nhỏ nhất trong mảng
int min = mmc[0];
for (int i = 1; i < dem; i++)
{
    if (min > mmc[i])
        min = mmc[i];
}
Console.WriteLine("Phan tu nho nhat trong mang la : {0}", min);
```

Bước 9. Chạy chương trình và ghi nhận kết quả.

Bước 10. Bổ sung đoạn mã sau để thực hiện thao tác tìm kiếm

```
// Tìm vị trí xuất hiện đầu tiên của một phần tử X
Console.Write("Nhap vao phan tu X can tim : ");
int x = int.Parse(Console.ReadLine());

int viTri = -1;
for (int i = 0; i < dem; i++)
{
    if (mmc[i] == x)
    {
        viTri = i;
        break;
    }
}

if (viTri == -1)
    Console.WriteLine("Khong tim thay phan tu {0}", x);
else
    Console.WriteLine("Vi tri cua {0} la {1}", x, viTri);
```

Bước 11. Chạy chương trình, ghi nhận lại kết quả ứng với các giá trị X khác nhau.

Bước 12. Tiếp tục bổ sung đoạn mã để sắp xếp các phần tử tăng dần như sau:

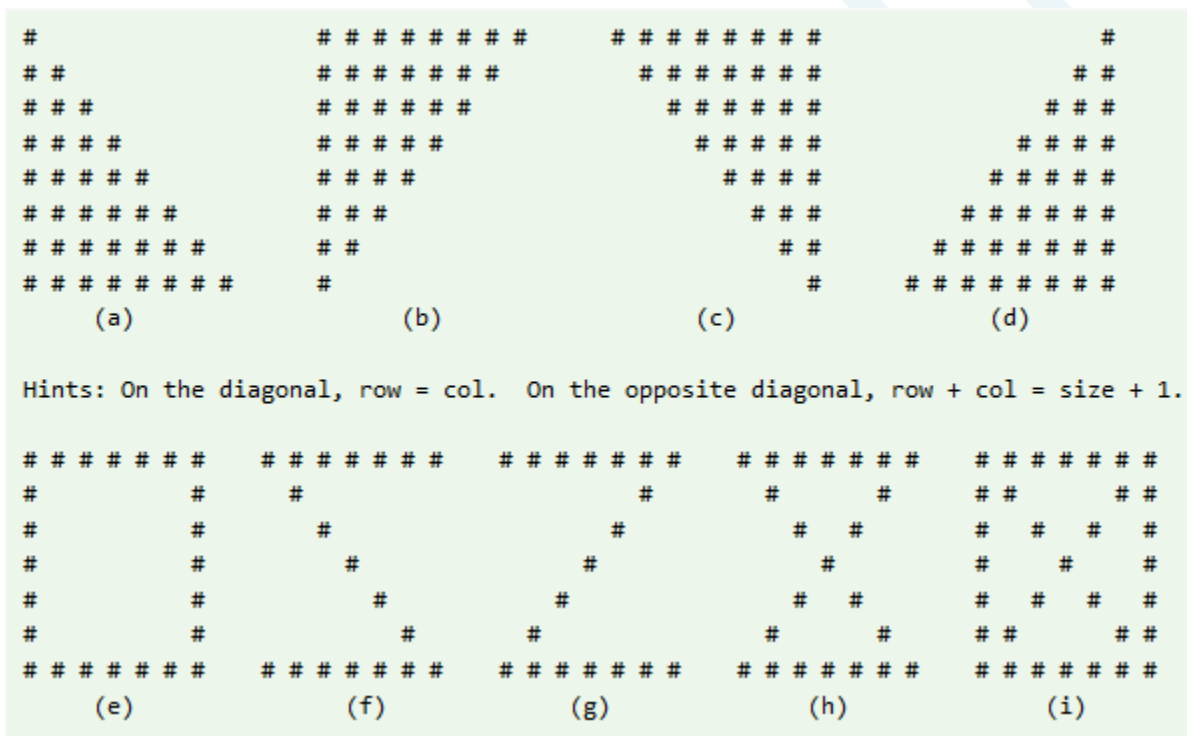
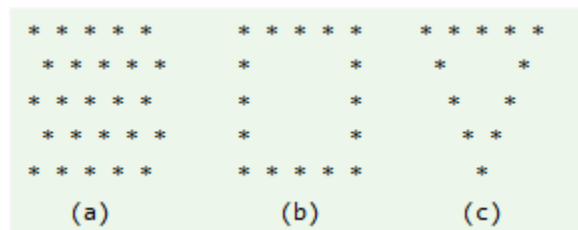
```
// Sắp xếp các phần tử của mảng theo thứ tự tăng dần
for (int i = 0; i < dem - 1; i++)
{
    for (int j = i + 1; j < dem; j++)
    {
        if (mmc[i] > mmc[j])
        {
            int tam = mmc[i];
            mmc[i] = mmc[j];
            mmc[j] = tam;
        }
    }
}
```

Bước 13. Sao chép đoạn mã xuất phần tử ở bước 5 vào sau đoạn mã trên

Bước 14. Chạy chương trình, ghi nhận kết quả và mô tả toàn bộ hoạt động của chương trình

## D. Bài tập bắt buộc

1. Viết chương trình vẽ các hình sau: (Kích thước của hình được nhập từ bàn phím)



2. Viết chương trình xuất các chuỗi có định dạng, canh lề như sau:

DANH SACH SINH VIEN			
MSSV	Ho va Ten	Khoa	Diem
1211520	Le Duy Tung	33	4.57
1210152	Tran Van Tien	34	UT
1211962	Pham Duc Anh	33	6.48
1211518	Nguyen Hong Phuc	36	UT
1211510	Pham Minh Duc	30	8.05
1211793	Nguyen Khac Uy	33	UT
1211519	Nguyen Tan Tai	34	2.89

3. Viết chương trình tính giá trị biểu thức và xuất giá trị các biến sau mỗi lệnh

- int x = 7, y = -3;
- int z = x++ + x++;

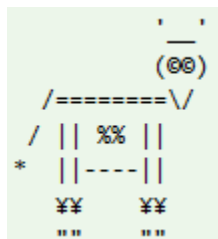
- c.  $z = x++ + x++ + ++x$ ;  
 d.  $z = ++x + y-- - ++y - x-- - x-- - ++y - x--$ ;  
 e.  $z = x++ + ++y - x-- + --y$ ;  
 f.  $z = x++ * y++ / ++x - --y \% x++$ ;  
 g.  $z = x++ + ++y - x-- + --y$ ;
- Viết chương trình nhập vào bán kính mặt đáy R và chiều cao H của hình trụ tròn. Sau đó xuất ra màn hình chu vi, diện tích mặt đáy, diện tích các mặt và thể tích hình trụ tròn.
  - Viết chương trình tính diện tích tam giác với độ dài 3 cạnh nhập từ bàn phím
  - Viết chương trình giải phương trình bậc 2 với các hệ số được nhập từ bàn phím
  - Viết chương trình cho phép nhập vào một số nguyên dương và xuất ra số đó ở dạng nhị phân, bát phân, thập lục phân.
  - Tìm n số nguyên tố đầu tiên.
  - Viết chương trình thực hiện các thao tác sau trên mảng một chiều chứa các số thực
    - Tìm và xuất các phần tử trong khoảng (min, max) với min, max nhập từ bàn phím
    - Tìm vị trí của tất cả các số âm và nằm ở vị trí lẻ
    - Tính tổng tất cả các số nằm ở vị trí chia hết cho 3
    - Tính trung bình cộng của các số trong mảng
    - Xuất ra k số lớn nhất trong mảng với k nhập từ bàn phím
  - Viết chương trình cho phép nhập vào ngày, tháng, năm. Sau đó xuất ra thứ trong tuần tương ứng ngày đó theo công thức sau:

$d = \text{ngày}, m = \text{tháng}, y = \text{năm}$   
 $y0 = y - (14 - m) / 12$   
 $x = y0 + y0/4 - y0/100 + y0/400$   
 $m0 = m + 12 * ((14 - m) / 12) - 2$   
 $d0 = (d + x + (31 * m0) / 12) \bmod 7$

với  $d0 = 0$  là Chủ nhật,  $d0 = 1$  là Thứ 2,  $d0 = 2$  là Thứ ba, ...

## E. Bài tập làm thêm

- Vẽ hình chú chó (Gợi ý: Mã ASCII của ¥ là 165 (00A5H) và © là 169 (00A9H))



- Xuất các tam giác sau:



```

      1
    1 2 1
  1 2 4 2 1
1 2 4 8 4 2 1
  1 2 4 8 16 8 4 2 1
    1 2 4 8 16 32 16 8 4 2 1
      1 2 4 8 16 32 64 32 16 8 4 2 1
        1 2 4 8 16 32 64 128 64 32 16 8 4 2 1

```

(a) PowerOf2Triangle

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1

```

(b) PascalTriangle1

```

      1
    1 1
  1 2 1
1 3 3 1
  1 4 6 4 1
    1 5 10 10 5 1
      1 6 15 20 15 6 1

```

(c) PascalTriangle2

```

1          1 2 3 4 5 6 7 8          1          8 7 6 5 4 3 2 1
1 2        1 2 3 4 5 6 7          2 1        7 6 5 4 3 2 1
1 2 3      1 2 3 4 5 6          3 2 1        6 5 4 3 2 1
1 2 3 4    1 2 3 4 5          4 3 2 1        5 4 3 2 1
1 2 3 4 5  1 2 3 4          5 4 3 2 1        4 3 2 1
1 2 3 4 5 6  1 2 3          6 5 4 3 2 1        3 2 1
1 2 3 4 5 6 7  1 2          7 6 5 4 3 2 1        2 1
1 2 3 4 5 6 7 8  1          8 7 6 5 4 3 2 1        1
(m)              (n)              (o)              (p)

```

```

      1
    1 2 1
  1 2 3 2 1
1 2 3 4 3 2 1
  1 2 3 4 5 4 3 2 1
    1 2 3 4 5 6 5 4 3 2 1
      1 2 3 4 5 6 7 6 5 4 3 2 1
        1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
(q)

```

```

1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
1 2 3 4 5 6 7 6 5 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
1 2 3 4 5 4 3 2 1
1 2 3 4 3 2 1
1 2 3 2 1
1 2 1
1
(r)

```

```

1          1          1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
1 2        2 1        1 2 3 4 5 6 7 7 6 5 4 3 2 1
1 2 3      3 2 1      1 2 3 4 5 6 6 5 4 3 2 1
1 2 3 4    4 3 2 1    1 2 3 4 5 5 4 3 2 1
1 2 3 4 5  5 4 3 2 1  1 2 3 4 4 3 2 1
1 2 3 4 5 6  6 5 4 3 2 1  1 2 3 3 2 1
1 2 3 4 5 6 7  7 6 5 4 3 2 1  1 2 2 1
1 2 3 4 5 6 7 8 7 6 5 4 3 2 1  1 1 1
(s)              (t)

```

3. Tính giai thừa của số nguyên dương  $n$
4. Xuất các số từ 1000 đến 2000, mỗi dòng có 5 số.
5. Nhập số nguyên dương  $n$ . Đếm số bit 1 trong biểu diễn nhị phân của  $n$ .
6. Tìm vị trí các số nguyên tố trong mảng
7. Đếm số lượng số hoàn chỉnh trong mảng
8. Viết chương trình tính tổng diện tích các mặt và thể tích của hình hộp chữ nhật.
9. Tìm ước chung lớn nhất và bội chung nhỏ nhất của 2 số nguyên  $a, b$ .
10. Số Fibonacci được định nghĩa như sau:  $F_0 = 0, F_1 = 1, F_n = F_{n-1} + F_{n-2}$ . Hãy viết chương trình xuất ra  $n$  số Fibonnaci đầu tiên với  $n$  được nhập từ bàn phím. Ví dụ với  $n = 10$ , ta có dãy 10 số như sau: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34