

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA CÔNG NGHỆ PHẦN MỀM



**MÔN LẬP TRÌNH TRỰC QUAN  
BÀI TẬP THỰC HÀNH 1**

GVHD: Nguyễn Ngọc Quý

Sinh viên thực hiện: Đặng Trường Sinh - 24521524

# IT008 – LẬP TRÌNH TRỰC QUAN

## NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

....., ngày ..... tháng ..... năm 2025

Người nhận xét

(Ký tên và ghi rõ họ tên)

**MỤC LỤC**

1. Viết chương trình cho phép tạo mảng một chiều gồm n số nguyên ngẫu nhiên .....	8
2. Viết chương trình nhập số nguyên dương n, tính tổng các số nguyên tố < n và xuất kết quả ra màn hình. ....	15
3. Viết chương trình nhập ngày tháng năm, cho biết ngày tháng năm đó có hợp lệ không? .....	18
4. Viết chương trình nhập vào tháng và năm. In ra số ngày của tháng đó .....	22
5. Viết chương trình nhập vào ngày tháng năm, cho biết thứ trong tuần. ....	27
6. Viết chương trình cho phép tạo ma trận chứa các số nguyên ngẫu nhiên gồm n dòng, m cột. Cài đặt hàm thực hiện các chức năng sau:.....	31
7. Link source code GitHub: .....	41

**DANH MỤC BẢNG**

Bảng 1.1 – Nội dung code của bài 1 .....	8
Bảng 1.2 – Testcase 1 của bài 1 .....	11
Bảng 1.3 – Testcase 2 của bài 1 .....	12
Bảng 1.4 - Testcase 3 của bài 1 .....	12
Bảng 1.5 - Testcase 4 của bài 1 .....	13
Bảng 1.6 - Testcase 5 của bài 1 .....	14
Bảng 2.1 – Nội dung code của bài 2 .....	15
Bảng 2.2 - Testcase 1 của bài 2 .....	16
Bảng 2.3 - Testcase 2 của bài 2 .....	17
Bảng 2.4 - Testcase 3 của bài 2 .....	17
Bảng 2.5 - Testcase 4 của bài 2 .....	17
Bảng 3.1 – Nội dung code của bài 3 .....	18
Bảng 3.2 - Testcase 1 của bài 3 .....	19
Bảng 3.3 - Testcase 2 của bài 3 .....	20
Bảng 3.4 - Testcase 3 của bài 3 .....	20
Bảng 3.5 - Testcase 4 của bài 3 .....	21
Bảng 3.6 - Testcase 5 của bài 3 .....	21
Bảng 3.7 - Testcase 6 của bài 3 .....	21
Bảng 3.8. Testcase 7 của bài 3 .....	22
Bảng 4.1 - Nội dung code của bài 4 .....	22
Bảng 4.2 - Testcase 1 của bài 4 .....	24
Bảng 4.3 - Testcase 2 của bài 4 .....	24
Bảng 4.4 - Testcase 3 của bài 4 .....	24
Bảng 4.5 - Testcase 4 của bài 4 .....	25
Bảng 4.6 - Testcase 5 của bài 4 .....	25
Bảng 4.7 - Testcase 6 của bài 4 .....	25
Bảng 4.8 - Testcase 7 của bài 4 .....	26
Bảng 4.9. Testcase 8 của bài 4 .....	26
Bảng 5.1. Nội dung code của bài 5 .....	27

## **IT008 – LẬP TRÌNH TRỰC QUAN**

Bảng 5.2 - Testcase 1 của bài 5 .....	28
Bảng 5.3 - Testcase 2 của bài 5 .....	29
Bảng 5.4 - Testcase 3 của bài 5 .....	29
Bảng 5.5 - Testcase 4 của bài 5 .....	30
Bảng 5.6 - Testcase 5 của bài 5 .....	30
Bảng 6.1 - Nội dung code của bài 6 .....	31
Bảng 6.2 - Testcase 1 của bài 6 .....	36
Bảng 6.3 - Testcase 2 của bài 6 .....	37
Bảng 6.4 - Testcase 3 của bài 6 .....	38
Bảng 6.5 - Testcase 4 của bài 6 .....	39

**DANH MỤC HÌNH ẢNH**

Hình 1. Testcase 1 của bài 1.....	11
Hình 2. Testcase 2 của bài 1.....	12
Hình 3. Testcase 3 của bài 1.....	13
Hình 4. Testcase 4 của bài 1.....	14
Hình 5. Testcase 5 của bài 1.....	14
Hình 6. Testcase 1 của bài 2.....	16
Hình 7. Testcase 2 của bài 2.....	17
Hình 8. Testcase 3 của bài 2.....	17
Hình 9. Testcase 4 của bài 2.....	18
Hình 10. Testcase 1 của bài 3.....	20
Hình 11. Testcase 2 của bài 3.....	20
Hình 12. Testcase 3 của bài 3.....	20
Hình 13. Testcase 4 của bài 3.....	21
Hình 14. Testcase 5 của bài 3.....	21
Hình 15. Testcase 6 của bài 3.....	22
Hình 16. Testcase 7 của bài 3.....	22
Hình 17. Testcase 1 của bài 4.....	24
Hình 18. Testcase 2 của bài 4.....	24
Hình 19. Testcase 3 của bài 4.....	25
Hình 20. Testcase 4 của bài 4.....	25
Hình 21. Testcase 5 của bài 4.....	25
Hình 22. Testcase 6 của bài 4.....	26
Hình 23. Testcase 7 của bài 4.....	26
Hình 24. Testcase 8 của bài 4.....	26
Hình 25. Testcase 1 của bài 5.....	29
Hình 26. Testcase 2 của bài 5.....	29
Hình 27. Testcase 3 của bài 5.....	30
Hình 28. Testcase 4 của bài 5.....	30
Hình 29. Testcase 5 của bài 5.....	30

## **IT008 – LẬP TRÌNH TRỰC QUAN**

Hình 30. Testcase 1 của bài 6.....	37
Hình 31. Testcase 2 của bài 6.....	38
Hình 32. Testcase 3 của bài 6.....	39
Hình 33. Testcase 4 của bài 6.....	41

## NỘI DUNG BÀI LÀM

### 1. Viết chương trình cho phép tạo mảng một chiều gồm n số nguyên ngẫu nhiên.

a. Tính tổng các số lẻ trong mảng

b. Đếm số nguyên tố trong mảng

c. Tìm số chính phương nhỏ nhất (nếu không có hàm trả về -1)

Mô tả đầu vào và đầu ra của bài toán:

- Input: Một mảng một chiều gồm n số nguyên ngẫu nhiên.
- Output: Kết quả tính tổng các số lẻ trong mảng, đếm số nguyên tố trong mảng và tìm số chính phương nhỏ nhất.

Nội dung code của Bài 1

*Bảng 1.1 – Nội dung code của bài 1.*

```

using System;
using System.Text;

namespace BTH1_Bai01
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;
            int n = 0;
            while (true)
            {
                Console.WriteLine("Nhập kích thước của mảng (n > 0): ");
                string input = Console.ReadLine();

                // Kiểm tra dữ liệu nhập vào
                if (!int.TryParse(input, out n))
                {
                    Console.WriteLine("Bạn phải nhập một số nguyên!");
                    continue;
                }

                if (n <= 0)
                {
                    Console.WriteLine("Kích thước mảng phải lớn hơn 0!");
                    continue;
                }

                break;
            }

            int[] a = new int[n];
            ArrayImport(a, n);
            Menu();
        }
    }
}

```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        Choose(a, n);
    }

    static void ArrayImport(int[] a, int n)
    {
        Random random = new Random();
        for (int i = 0; i < n; i++)
            a[i] = random.Next(-100, 100);
        Console.WriteLine("Mảng vừa tạo: ");
        for (int i = 0; i < n; i++)
            Console.Write(a[i] + " ");
        Console.WriteLine();
    }

    // Tính tổng các số lẻ trong mảng
    static int OddTotal(int[] a, int n)
    {
        int sum = 0;
        for (int i = 0; i < n; i++)
        {
            if (a[i] % 2 != 0)
                sum += a[i];
        }
        return sum;
    }

    // Đếm số nguyên tố có trong mảng
    static int CountPrime(int[] a, int n)
    {
        bool isPrime(int x)
        {
            if (x < 2) return false;
            for (int i = 2; i <= Math.Sqrt(x); i++)
            {
                if (x % i == 0) return false;
            }
            return true;
        }

        int count = 0;
        for (int i = 0; i < n; i++)
        {
            if (isPrime(a[i]))
                count++;
        }
        return count;
    }

    // Tìm số chính phương nhỏ nhất trong mảng
    static int FindMinSquareNumber(int[] a, int n)
    {
        bool isSquareNumber(int x)
        {
            if (x < 0) return false;
            int temp = (int)Math.Sqrt(x);
            return temp * temp == x;
        }

        int min = int.MaxValue;
        bool found = false;

        for (int i = 0; i < n; i++)
        {
            if (isSquareNumber(a[i]) && a[i] < min)
            {

```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        min = a[i];
        found = true;
    }

    return found ? min : -1;
}

static void Menu()
{
    Console.WriteLine("\n-----MENU-----");
    Console.WriteLine("1. Tính tổng các số lẻ trong mảng");
    Console.WriteLine("2. Đếm số nguyên tố trong mảng");
    Console.WriteLine("3. Tìm số chính phương nhỏ nhất trong mảng");
    Console.WriteLine("4. Thoát!");
}

static void Choose(int[] a, int n)
{
    Console.Write("\nNhập lựa chọn của bạn: ");
    string input = Console.ReadLine();
    int choice;

    // Kiểm tra lựa chọn hợp lệ
    if (!int.TryParse(input, out choice))
    {
        Console.WriteLine("Bạn phải nhập số nguyên trong menu!");
        Choose(a, n);
        return;
    }

    switch (choice)
    {
        case 1:
            Console.WriteLine("Tổng các số lẻ trong mảng: " +
OddTotal(a, n));
            Choose(a, n);
            break;
        case 2:
            Console.WriteLine("Số lượng các số nguyên tố trong mảng: " +
+ CountPrime(a, n));
            Choose(a, n);
            break;
        case 3:
            int minSquare = FindMinSquareNumber(a, n);
            if (minSquare == -1)
                Console.WriteLine("Không có số chính phương trong
mảng.");
            else
                Console.WriteLine("Số chính phương nhỏ nhất trong mảng:
" + minSquare);
            Choose(a, n);
            break;
        case 4:
            Console.WriteLine("Đã thoát chương trình!");
            return;
        default:
            Console.WriteLine("Lựa chọn không hợp lệ!");
            Choose(a, n);
            break;
    }
}
}
```

Các Testcase của chương trình:

Bảng 1.2 – Testcase 1 của bài 1

Input:	Output:
5	Mảng vừa tạo: -21 38 1 -20 1
1	Tổng các số lẻ trong mảng: -19
2	Số lượng các số nguyên tố trong mảng: 0
3	Số chính phương nhỏ nhất trong mảng: 1
4	Đã thoát chương trình!

```
Nhập kích thước của mảng (n > 0): 5
Mảng vừa tạo: -21 38 1 -20 1

-----MENU-----
1. Tính tổng các số lẻ trong mảng
2. Đếm số nguyên tố trong mảng
3. Tìm số chính phương nhỏ nhất trong mảng
4. Thoát!

Nhập lựa chọn của bạn: 1
Tổng các số lẻ trong mảng: -19

Nhập lựa chọn của bạn: 2
Số lượng các số nguyên tố trong mảng: 0

Nhập lựa chọn của bạn: 3
Số chính phương nhỏ nhất trong mảng: 1

Nhập lựa chọn của bạn: 4
Đã thoát chương trình!

D:\Study\IT008_L?p trình tr?c quan\Th?c hành\BTH1_DangTruongSinh_24521524\B
8) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Optio
le when debugging stops.
Press any key to close this window . . .|
```

Hình 1. Testcase 1 của bài 1

## IT008 – LẬP TRÌNH TRỰC QUAN

Bảng 1.3 – Testcase 2 của bài 1

Input:	Output:
0	Kích thước mảng phải lớn hơn 0!
3	Mảng vừa tạo: 78 -71 -20
1	Tổng các số lẻ trong mảng: -71
4	Đã thoát chương trình!

```
Nhập kích thước của mảng (n > 0): 0
Kích thước mảng phải lớn hơn 0!
Nhập kích thước của mảng (n > 0): 3
Mảng vừa tạo: 78 -71 -20

-----MENU-----
1. Tính tổng các số lẻ trong mảng
2. Đếm số nguyên tố trong mảng
3. Tìm số chính phương nhỏ nhất trong mảng
4. Thoát!

Nhập lựa chọn của bạn: 1
Tổng các số lẻ trong mảng: -71

Nhập lựa chọn của bạn: 4
Đã thoát chương trình!
```

Hình 2. Testcase 2 của bài 1

Bảng 1.4 - Testcase 3 của bài 1

Input:	Output:
-1	Kích thước mảng phải lớn hơn 0!
4	Mảng vừa tạo: 43 -32 -69 -99
5	Lựa chọn không hợp lệ!
0	Lựa chọn không hợp lệ!
gjkghk	Lựa chọn không hợp lệ!
4	Đã thoát chương trình!

## IT008 – LẬP TRÌNH TRỰC QUAN

```
Nhập kích thước của mảng (n > 0): -1
Kích thước mảng phải lớn hơn 0!
Nhập kích thước của mảng (n > 0): 4
Mảng vừa tạo: 43 -32 -69 -99

-----MENU-----
1. Tính tổng các số lẻ trong mảng
2. Đếm số nguyên tố trong mảng
3. Tìm số chính phương nhỏ nhất trong mảng
4. Thoát!

Nhập lựa chọn của bạn: 5
Lựa chọn không hợp lệ!

Nhập lựa chọn của bạn: 0
Lựa chọn không hợp lệ!

Nhập lựa chọn của bạn: gjkghk
Bạn phải nhập số nguyên trong menu!

Nhập lựa chọn của bạn: 4
Đã thoát chương trình!
```

Hình 3. Testcase 3 của bài 1

Bảng 1.5 - Testcase 4 của bài 1

Input:	Output:
0	Kích thước mảng phải lớn hơn 0!
1.5	Bạn phải nhập một số nguyên!
4	Mảng vừa tạo: -58 77 -65 50
←	Bạn phải nhập số nguyên trong menu!
1.5	Bạn phải nhập số nguyên trong menu!
1	Tổng các số lẻ trong mảng: 12
4	Đã thoát chương trình!

## IT008 – LẬP TRÌNH TRỰC QUAN

```
Nhập kích thước của mảng (n > 0): 0
Kích thước mảng phải lớn hơn 0!
Nhập kích thước của mảng (n > 0): 1.5
Bạn phải nhập một số nguyên!
Nhập kích thước của mảng (n > 0): 4
Mảng vừa tạo: -58 77 -65 50

-----MENU-----
1. Tính tổng các số lẻ trong mảng
2. Đếm số nguyên tố trong mảng
3. Tìm số chính phương nhỏ nhất trong mảng
4. Thoát!

Nhập lựa chọn của bạn:
Bạn phải nhập số nguyên trong menu!

Nhập lựa chọn của bạn: 1.5
Bạn phải nhập số nguyên trong menu!

Nhập lựa chọn của bạn: 1
Tổng các số lẻ trong mảng: 12

Nhập lựa chọn của bạn: 4
Đã thoát chương trình!
```

Hình 4. Testcase 4 của bài I

Bảng 1.6 - Testcase 5 của bài I

Input:	Output:
-1.2	Bạn phải nhập một số nguyên!
ba	Bạn phải nhập một số nguyên!
0	Kích thước mảng phải lớn hơn 0!
-3	Kích thước mảng phải lớn hơn 0!
3	Mảng vừa tạo: 54 88 -42
←	Bạn phải nhập số nguyên trong menu!
2	Số lượng các số nguyên tố trong mảng: 0
4	Đã thoát chương trình!

```
Nhập kích thước của mảng (n > 0): -1.2
Bạn phải nhập một số nguyên!
Nhập kích thước của mảng (n > 0): ba
Bạn phải nhập một số nguyên!
Nhập kích thước của mảng (n > 0): 0
Kích thước mảng phải lớn hơn 0!
Nhập kích thước của mảng (n > 0): -3
Kích thước mảng phải lớn hơn 0!
Nhập kích thước của mảng (n > 0): 3
Mảng vừa tạo: 54 88 -42

-----MENU-----
1. Tính tổng các số lẻ trong mảng
2. Đếm số nguyên tố trong mảng
3. Tìm số chính phương nhỏ nhất trong mảng
4. Thoát!

Nhập lựa chọn của bạn:
Bạn phải nhập số nguyên trong menu!

Nhập lựa chọn của bạn: 2
Số lượng các số nguyên tố trong mảng: 0

Nhập lựa chọn của bạn: 4
Đã thoát chương trình!
```

Hình 5. Testcase 5 của bài I

## 2. Viết chương trình nhập số nguyên dương n, tính tổng các số nguyên tố < n và xuất kết quả ra màn hình.

Mô tả đầu vào và đầu ra của bài toán:

- Input: Nhập vào 1 số nguyên dương n.
- Output: Kết quả tính tổng các số nguyên tố < n và xuất kết quả ra màn hình.

Nội dung code của bài 2:

Bảng 2.1 – Nội dung code của bài 2

```

using System;
using System.Text;

namespace BTH1_Bai02
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;
            int n;

            while (true)
            {
                Console.Write("Nhập n (n > 0): ");
                string input = Console.ReadLine();

                if (string.IsNullOrWhiteSpace(input))
                {
                    Console.WriteLine("Bạn chưa nhập gì, vui lòng nhập một
số nguyên dương!");
                    continue;
                }

                if (double.TryParse(input, out double doubleValue))
                {
                    if (doubleValue % 1 != 0)
                    {
                        Console.WriteLine("Bạn đã nhập số thập phân ({0}),
vui lòng nhập số nguyên dương!", doubleValue);
                        continue;
                    }
                }

                if (int.TryParse(input, out n))
                {
                    if (n <= 0)
                    {
                        Console.WriteLine("Số bạn nhập là {0} (<= 0), vui
lòng nhập số nguyên dương!", n);
                        continue;
                    }
                    break;
                }
                else
                {
                    Console.WriteLine("Giá trị bạn nhập không hợp lệ, vui
lòng nhập số nguyên dương!");
                }
            }
        }
    }
}

```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        }

        int sum = PrintAndSumPrimes(n);
        Console.WriteLine("\nTổng các số nguyên tố < {0}: {1}", n, sum);
    }

    static bool isPrime(int x)
    {
        if (x < 2) return false;
        for (int i = 2; i <= Math.Sqrt(x); i++)
        {
            if (x % i == 0) return false;
        }
        return true;
    }

    static int PrintAndSumPrimes(int n)
    {
        int sum = 0;
        Console.Write("Các số nguyên tố < {0}: ", n);
        for (int i = 2; i < n; i++)
        {
            if (isPrime(i))
            {
                Console.Write(i + " ");
                sum += i;
            }
        }
        return sum;
    }
}
```

Bảng 2.2 - Testcase 1 của bài 2

Input:	Output
7	Các số nguyên tố < 7: 2 3 5 Tổng các số nguyên tố < 7: 10

```
Nhập n (n > 0): 7
Các số nguyên tố < 7: 2 3 5
Tổng các số nguyên tố < 7: 10
```

Hình 6. Testcase 1 của bài 2

## IT008 – LẬP TRÌNH TRỰC QUAN

Bảng 2.3 - Testcase 2 của bài 2

Input:	Output:
93	Các số nguyên tố < 93: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 Tổng các số nguyên tố < 93: 963

```
Nhập n (n > 0): 93
Các số nguyên tố < 93: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67
71 73 79 83 89
Tổng các số nguyên tố < 93: 963
```

Hình 7. Testcase 2 của bài 2

Bảng 2.4 - Testcase 3 của bài 2

Input:	Output:
←	Bạn chưa nhập gì, vui lòng nhập một số nguyên dương!
-3	Số bạn nhập là -3 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
13	Các số nguyên tố < 13: 2 3 5 7 11 Tổng các số nguyên tố < 13: 28

```
Nhập n (n > 0):
Bạn chưa nhập gì, vui lòng nhập một số nguyên dương!
Nhập n (n > 0): -3
Số bạn nhập là -3 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
Nhập n (n > 0): 13
Các số nguyên tố < 13: 2 3 5 7 11
Tổng các số nguyên tố < 13: 28
```

Hình 8. Testcase 3 của bài 2

Bảng 2.5 - Testcase 4 của bài 2

Input:	Output:
-2	Số bạn nhập là -2 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
1.2	Bạn đã nhập số thập phân (1.2), vui lòng nhập số nguyên dương!
0	Số bạn nhập là 0 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
1	Các số nguyên tố < 1: Tổng các số nguyên tố < 1: 0

```
Nhập n (n > 0): -2
Số bạn nhập là -2 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
Nhập n (n > 0): 1.2
Bạn đã nhập số thập phân (1.2), vui lòng nhập số nguyên dương!
Nhập n (n > 0): 0
Số bạn nhập là 0 ( $\leq 0$ ), vui lòng nhập số nguyên dương!
Nhập n (n > 0): 1
Các số nguyên tố < 1:
Tổng các số nguyên tố < 1: 0
```

Hình 9. Testcase 4 của bài 2

### 3. Viết chương trình nhập ngày tháng năm, cho biết ngày tháng năm đó có hợp lệ không?

Mô tả đầu vào và đầu ra của bài toán:

- Input: Nhập vào 3 số nguyên lần lượt là ngày, tháng, năm.
- Output: Kiểm tra xem ngày, tháng, năm vừa nhập có hợp lệ hay không.

Nội dung code của bài 3:

Bảng 3.1 – Nội dung code của bài 3.

```
using System;
using System.Text;

namespace BTH1_Bai03
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;

            int day = NhapSoNguyen("ngày");
            int month = NhapSoNguyen("tháng");
            int year = NhapSoNguyen("năm");

            if (IsValidDay(day, month, year))
                Console.WriteLine($"{day}/{month}/{year} là ngày hợp lệ");
            else
                Console.WriteLine($"{day}/{month}/{year} không phải là ngày
hợp lệ");
        }

        static int NhapSoNguyen(string tenTruong)
        {
            int value;
            while (true)
            {
                Console.Write($"Nhập {tenTruong}: ");
                string input = Console.ReadLine();
                if (int.TryParse(input, out value) && value > 0)
                    break;
                else
                    Console.WriteLine("Nhập số nguyên dương!");
            }
            return value;
        }
    }
}
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```

        if (string.IsNullOrWhiteSpace(input))
        {
            Console.WriteLine($"Bạn chưa nhập {tenTruong}, vui lòng
nhập lại!");
            continue;
        }

        if (double.TryParse(input, out double dbl))
        {
            if (dbl % 1 != 0)
            {
                Console.WriteLine($"Bạn đã nhập số thập phân
({dbl}), vui lòng nhập số nguyên!");
                continue;
            }
        }

        if (int.TryParse(input, out value))
        {
            break;
        }
        else
        {
            Console.WriteLine($"Giá trị '{input}' không hợp lệ, vui
lòng nhập số nguyên!");
        }
    }
    return value;
}

static bool IsValidDay(int day, int month, int year)
{
    if (year <= 0) return false;
    if (month < 1 || month > 12) return false;
    if (day < 1) return false;

    int[] DayInMonth =
    {
        31,
        DateTime.IsLeapYear(year) ? 29 : 28, 31, 30, 31, 30, 31, 31,
30, 31, 30, 31
    };

    if (day > DayInMonth[month - 1])
        return false;

    return true;
}
}

```

Bảng 3.2 - Testcase 1 của bài 3

Input:	Output:
16	16/10/2025 là ngày hợp lệ
10	
2025	

Nhập ngày: 16  
Nhập tháng: 10  
Nhập năm: 2025  
16/10/2025 là ngày hợp lệ

Hình 10. Testcase 1 của bài 3

Bảng 3.3 - Testcase 2 của bài 3

Input:	Output:
29 2 2024	29/2/2024 là ngày hợp lệ

Nhập ngày: 29  
Nhập tháng: 2  
Nhập năm: 2024  
29/2/2024 là ngày hợp lệ

Hình 11. Testcase 2 của bài 3

Bảng 3.4 - Testcase 3 của bài 3

Input:	Output:
29 2 2023	29/2/2023 không phải là ngày hợp lệ

Nhập ngày: 29  
Nhập tháng: 2  
Nhập năm: 2023  
29/2/2023 không phải là ngày hợp lệ

Hình 12. Testcase 3 của bài 3

## IT008 – LẬP TRÌNH TRỰC QUAN

Bảng 3.5 - Testcase 4 của bài 3

Input:	Output:
abc	Giá trị 'abc' không hợp lệ, vui lòng nhập số nguyên!
23	23/9/2025 là ngày hợp lệ
9	
2025	

```
Nhập ngày: abc
Giá trị 'abc' không hợp lệ, vui lòng nhập số nguyên!
Nhập ngày: 23
Nhập tháng: 9
Nhập năm: 2025
23/9/2025 là ngày hợp lệ
```

Hình 13. Testcase 4 của bài 3

Bảng 3.6 - Testcase 5 của bài 3

Input:	Output:
3.5	Bạn đã nhập số thập phân (3.5), vui lòng nhập số nguyên!
4	4/4/2023 là ngày hợp lệ
4	
2023	

```
Nhập ngày: 3.5
Bạn đã nhập số thập phân (3.5), vui lòng nhập số nguyên!
Nhập ngày: 4
Nhập tháng: 4
Nhập năm: 2023
4/4/2023 là ngày hợp lệ
```

Hình 14. Testcase 5 của bài 3

Bảng 3.7 - Testcase 6 của bài 3

Input:	Output:
←	Bạn chưa nhập ngày, vui lòng nhập lại!
11	11/-9/2001 không phải là ngày hợp lệ
-9	
2001	

```
Nhập ngày:  
Bạn chưa nhập ngày, vui lòng nhập lại!  
Nhập ngày: 11  
Nhập tháng: -9  
Nhập năm: 2001  
11/-9/2001 không phải là ngày hợp lệ
```

Hình 15. Testcase 6 của bài 3

Bảng 3.8. Testcase 7 của bài 3

Input:	Output:
31 13 2025	31/13/2025 không phải là ngày hợp lệ

```
Nhập ngày: 31  
Nhập tháng: 13  
Nhập năm: 2025  
31/13/2025 không phải là ngày hợp lệ
```

Hình 16. Testcase 7 của bài 3

#### 4. Viết chương trình nhập vào tháng và năm. In ra số ngày của tháng đó

Mô tả đầu vào và đầu ra của bài toán:

- Input: Nhập vào 2 số nguyên lần lượt là tháng, năm.
- Output: Kiểm tra xem tháng, năm vừa nhập có hợp lệ hay không. Nếu hợp lệ thì in ra số ngày của tháng đó.

Nội dung code của bài 4:

Bảng 4.1 - Nội dung code của bài 4

```
using System;  
using System.Text;
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
namespace BTH1_Bai04
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.UTF8;

            int month = NhapSoNguyen("tháng");
            int year = NhapSoNguyen("năm");

            if (IsValidMonth(month) && IsValidYear(year))
            {
                int days = DateTime.DaysInMonth(year, month);
                Console.WriteLine($"Tháng {month}/{year} có {days} ngày.");
            }
            else
            {
                Console.WriteLine($"Tháng {month}/{year} không hợp lệ!");
            }
        }

        static int NhapSoNguyen(string tenTruong)
        {
            int value;
            while (true)
            {
                Console.Write($"Nhập {tenTruong}: ");
                string input = Console.ReadLine();

                if (string.IsNullOrWhiteSpace(input))
                {
                    Console.WriteLine($"Bạn chưa nhập {tenTruong}, vui lòng
nhập lại!");
                    continue;
                }

                if (double.TryParse(input, out double dbl))
                {
                    if (dbl % 1 != 0)
                    {
                        Console.WriteLine($"Bạn đã nhập số thập phân
({dbl}), vui lòng nhập số nguyên!");
                        continue;
                    }
                }

                if (int.TryParse(input, out value))
                {
                    break;
                }
                else
                {
                    Console.WriteLine($"Giá trị '{input}' không hợp lệ, vui
lòng nhập số nguyên!");
                }
            }
            return value;
        }

        static bool IsValidMonth(int month)
        {
            return month >= 1 && month <= 12;
        }

        static bool IsValidYear(int year)
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
{  
    return year >= 1 && year <= 9999;  
}  
}
```

Bảng 4.2 - Testcase 1 của bài 4

Input:	Output:
10 2025	Tháng 10/2025 có 31 ngày.

```
Nhập tháng: 10  
Nhập năm: 2025  
Tháng 10/2025 có 31 ngày.
```

Hình 17. Testcase 1 của bài 4

Bảng 4.3 - Testcase 2 của bài 4

Input:	Output:
2 2024	Tháng 2/2024 có 29 ngày.

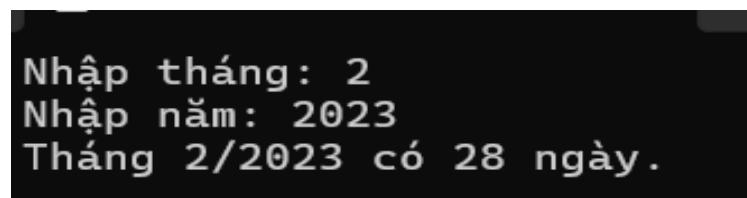
```
Nhập tháng: 2  
Nhập năm: 2024  
Tháng 2/2024 có 29 ngày.
```

Hình 18. Testcase 2 của bài 4

Bảng 4.4 - Testcase 3 của bài 4

Input:	Output:
2 2023	Tháng 2/2023 có 28 ngày

## IT008 – LẬP TRÌNH TRỰC QUAN

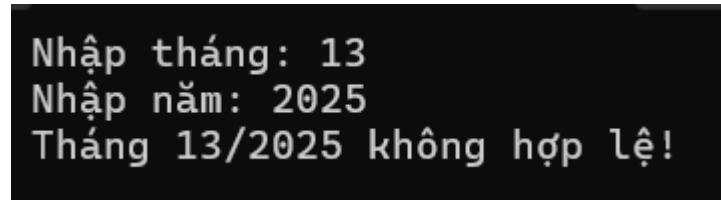


Nhập tháng: 2  
Nhập năm: 2023  
Tháng 2/2023 có 28 ngày.

Hình 19. Testcase 3 của bài 4

Bảng 4.5 - Testcase 4 của bài 4

Input:	Output:
13 2025	Tháng 13/2025 không hợp lệ!

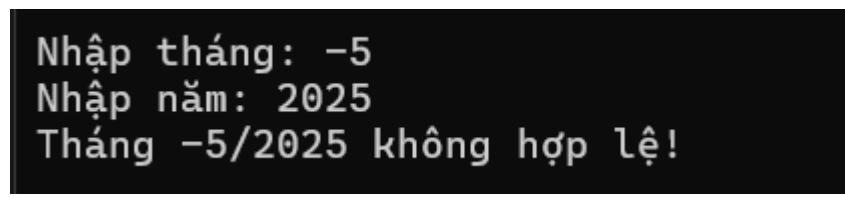


Nhập tháng: 13  
Nhập năm: 2025  
Tháng 13/2025 không hợp lệ!

Hình 20. Testcase 4 của bài 4

Bảng 4.6 - Testcase 5 của bài 4

Input:	Output:
-5 2025	Tháng -5/2025 không hợp lệ!



Nhập tháng: -5  
Nhập năm: 2025  
Tháng -5/2025 không hợp lệ!

Hình 21. Testcase 5 của bài 4

Bảng 4.7 - Testcase 6 của bài 4

Input:	Output:
5 0	Tháng 5/0 không hợp lệ!

## IT008 – LẬP TRÌNH TRỰC QUAN

```
Nhập tháng: 5  
Nhập năm: 0  
Tháng 5/0 không hợp lệ!
```

Hình 22. Testcase 6 của bài 4

Bảng 4.8 - Testcase 7 của bài 4

Input:	Output:
abc	Giá trị 'abc' không hợp lệ, vui lòng nhập số nguyên!
3	Tháng 3/-9999999 không hợp lệ!
-9999999	

```
Nhập tháng: abc  
Giá trị 'abc' không hợp lệ, vui lòng nhập số nguyên!  
Nhập tháng: 3  
Nhập năm: -9999999  
Tháng 3/-9999999 không hợp lệ!
```

Hình 23. Testcase 7 của bài 4

Bảng 4.9. Testcase 8 của bài 4

Input:	Output:
←	Bạn chưa nhập tháng, vui lòng nhập lại!
3.2	Bạn đã nhập số thập phân (3.2), vui lòng nhập số nguyên!
2	
4.7	Bạn đã nhập số thập phân (4.7), vui lòng nhập số nguyên!
←	Bạn chưa nhập năm, vui lòng nhập lại!
2006	Tháng 2/2006 có 28 ngày.

```
Nhập tháng:  
Bạn chưa nhập tháng, vui lòng nhập lại!  
Nhập tháng: 3.2  
Bạn đã nhập số thập phân (3.2), vui lòng nhập số nguyên!  
Nhập tháng: 2  
Nhập năm: 4.7  
Bạn đã nhập số thập phân (4.7), vui lòng nhập số nguyên!  
Nhập năm:  
Bạn chưa nhập năm, vui lòng nhập lại!  
Nhập năm: 2006  
Tháng 2/2006 có 28 ngày.
```

Hình 24. Testcase 8 của bài 4

## 5. Viết chương trình nhập vào ngày tháng năm, cho biết thứ trong tuần.

Mô tả đầu vào và đầu ra của bài toán:

- Input: Nhập vào 3 số nguyên lần lượt là ngày, tháng, năm.
- Output: Kiểm tra xem tháng, năm vừa nhập có hợp lệ hay không. Nếu hợp lệ thì cho biết đó là thứ mấy trong tuần.

Nội dung code của bài 5:

Bảng 5.1. Nội dung code của bài 5

```

using System;
using System.Text;
namespace BTH1_Bai05
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.UTF8;
            int day = NhapSoNguyen("ngày");
            int month = NhapSoNguyen("tháng");
            int year = NhapSoNguyen("năm");
            if (IsValidDay(day, month, year))
            {
                DateTime dateTime = new DateTime(year, month, day);
                Console.WriteLine($"{day}/{month}/{year} là
{getDayOfWeek(dateTime.DayOfWeek)}");
            }
            else
                Console.WriteLine($"{day}/{month}/{year} không hợp lệ");
        }

        static int NhapSoNguyen(string tenTruong)
        {
            int value;
            while (true)
            {
                Console.Write($"Nhập {tenTruong}: ");
                string input = Console.ReadLine();

                if (string.IsNullOrWhiteSpace(input))
                {
                    Console.WriteLine($"Bạn chưa nhập {tenTruong}, vui lòng
nhập lại!");
                    continue;
                }

                if (double.TryParse(input, out double dbl))
                {
                    if (dbl % 1 != 0)
                    {
                        Console.WriteLine($"Bạn đã nhập số thập phân ({dbl}),
vui lòng nhập số nguyên!");
                        continue;
                    }
                }
            }
        }
    }
}

```

```

        if (int.TryParse(input, out value))
        {
            break;
        }
        else
        {
            Console.WriteLine($"Giá trị '{input}' không hợp lệ, vui
lòng nhập số nguyên!");
        }
    }
    return value;
}

// Kiểm tra ngày có hợp lệ
static bool IsValidDay(int day, int month, int year)
{
    if (day < 1) return false;
    if (year < 1 || year > 9999) return false;
    if (month < 1 || month > 12) return false;
    int[] DayInMonth = { 31, DateTime.IsLeapYear(year) ? 29 : 28, 31,
30, 31, 30, 31, 30, 31 };
    if (day > DayInMonth[month - 1]) return false;
    return true;
}

// Chuyển thứ trong tuần sang tiếng Việt
static string getDayOfWeek(DayOfWeek date)
{
    int dayOfWeek = (int) date;
    switch (dayOfWeek)
    {
        case 0:
            return "Chủ Nhật";
        case 1:
            return "Thứ Hai";
        case 2:
            return "Thứ Ba";
        case 3:
            return "Thứ Tư";
        case 4:
            return "Thứ Năm";
        case 5:
            return "Thứ Sáu";
        case 6:
            return "Thứ Bảy";
        default: return "";
    }
}
}

```

Bảng 5.2 - Testcase 1 của bài 5

<b>Input:</b>	<b>Output:</b>
16	16/10/2025 là Thứ Năm
10	
2025	

## IT008 – LẬP TRÌNH TRỰC QUAN

```
Nhập ngày: 16  
Nhập tháng: 10  
Nhập năm: 2025  
16/10/2025 là Thứ Năm
```

Hình 25. Testcase 1 của bài 5

Bảng 5.3 - Testcase 2 của bài 5

Input:	Output:
←	Bạn chưa nhập ngày, vui lòng nhập lại!
3.4	Bạn đã nhập số thập phân (3.4), vui lòng nhập số nguyên!
9/	Giá trị '9/' không hợp lệ, vui lòng nhập số nguyên!
9	Giá trị 'hai' không hợp lệ, vui lòng nhập số nguyên!
hai	9/2/2006 là Thứ Năm
2	
2006	

```
Nhập ngày:  
Bạn chưa nhập ngày, vui lòng nhập lại!  
Nhập ngày: 3.4  
Bạn đã nhập số thập phân (3.4), vui lòng nhập số nguyên!  
Nhập ngày: 9/  
Giá trị '9/' không hợp lệ, vui lòng nhập số nguyên!  
Nhập ngày: 9  
Nhập tháng: hai  
Giá trị 'hai' không hợp lệ, vui lòng nhập số nguyên!  
Nhập tháng: 2  
Nhập năm: 2006  
9/2/2006 là Thứ Năm
```

Hình 26. Testcase 2 của bài 5

Bảng 5.4 - Testcase 3 của bài 5

Input:	Output:
29	Giá trị '/' không hợp lệ, vui lòng nhập số nguyên!
2	Giá trị '/2023' không hợp lệ, vui lòng nhập số nguyên!
/	29/2/2023 không hợp lệ
/2023	
2023	

## IT008 – LẬP TRÌNH TRỰC QUAN

```
Nhập ngày: 29
Nhập tháng: 2
Nhập năm: /
Giá trị '/' không hợp lệ, vui lòng nhập số nguyên!
Nhập năm: /2023
Giá trị '/2023' không hợp lệ, vui lòng nhập số nguyên!
Nhập năm: 2023
29/2/2023 không hợp lệ
```

Hình 27. Testcase 3 của bài 5

Bảng 5.5 - Testcase 4 của bài 5

Input:	Output:
-5	-5/2/205 không hợp lệ
2	
205	

```
Nhập ngày: -5
Nhập tháng: 2
Nhập năm: 205
-5/2/205 không hợp lệ
```

Hình 28. Testcase 4 của bài 5

Bảng 5.6 - Testcase 5 của bài 5

Input:	Output:
16	16/10/-2025 không hợp lệ
10	
-2025	

```
Nhập ngày: 16
Nhập tháng: 10
Nhập năm: -2025
16/10/-2025 không hợp lệ
```

Hình 29. Testcase 5 của bài 5

## 6. Viết chương trình cho phép tạo ma trận chứa các số nguyên ngẫu nhiên gồm n dòng, m cột. Cài đặt hàm thực hiện các chức năng sau:

- a. Xuất ma trận
- b. Tìm phần tử lớn nhất/nhỏ nhất
- c. Tìm dòng có tổng lớn nhất
- d. Tính tổng các số không phải là số nguyên tố
- e. Xóa dòng thứ k trong ma trận
- f. Xóa cột chứa phần tử lớn nhất trong ma trận

Mô tả đầu vào và đầu ra của bài toán:

- Input: Nhập vào 2 số nguyên lần lượt là số dòng và số cột của ma trận
- Output: Thực hiện các chức năng
  - o Xuất ma trận
  - o Tìm phần tử lớn nhất/nhỏ nhất
  - o Tìm dòng có tổng lớn nhất
  - o Tính tổng các số không phải là số nguyên tố
  - o Xóa dòng thứ k trong ma trận
  - o Xóa cột chứa phần tử lớn nhất trong ma trận

Nội dung code của bài 6:

Bảng 6.1 - Nội dung code của bài 6

```
using System;
using System.Text;
using System.Globalization;

namespace BTH1_Bai06
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.UTF8;
            Console.WriteLine("==== NHẬP KÍCH THƯỚC CỦA MA TRẬN ====");

            int n = NhapSoNguyen("số dòng", 1);
            int m = NhapSoNguyen("số cột", 1);

            int[,] matrix = new int[n, m];
            ImportMatrix(matrix, n, m);
            Choose(matrix, ref n, ref m);
        }

        //HÀM NHẬP CHUNG

        static int NhapSoNguyen(string tenTruong, int minValue = int.MinValue)
        {
            int value;
            while (true)
            {
                Console.Write($"Nhập {tenTruong}: ");
                if (int.TryParse(Console.ReadLine(), out value) && value > minValue)
                    break;
            }
            return value;
        }
    }
}
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        string input = Console.ReadLine();

        // Kiểm tra rỗng
        if (string.IsNullOrWhiteSpace(input))
        {
            Console.WriteLine($"Bạn chưa nhập {tenTruong}, vui lòng
nhập lại!");
            continue;
        }

        // Kiểm tra số thực
        if (double.TryParse(input, NumberStyles.Float,
CultureInfo.InvariantCulture, out double dbl))
        {
            if (dbl % 1 != 0)
            {
                Console.WriteLine($"Bạn đã nhập số thập phân ({dbl}),
vui lòng nhập số nguyên!");
                continue;
            }
        }

        // Kiểm tra số nguyên
        if (int.TryParse(input, out value))
        {
            if (value < minValue)
            {
                Console.WriteLine($"{tenTruong} phải ≥ {minValue}, vui
lòng nhập lại!");
                continue;
            }
            return value;
        }
        else
        {
            Console.WriteLine($"Giá trị '{input}' không hợp lệ, vui
lòng nhập số nguyên!");
        }
    }

    //KIỂM TRA MA TRẬN RỖNG
    static bool IsMatrixEmpty(int n, int m)
    {
        return n <= 0 || m <= 0;
    }

    static void Menu()
    {
        Console.WriteLine("\t----- MENU -----");
        Console.WriteLine("1. Xuất ma trận.");
        Console.WriteLine("2. Phần tử lớn nhất / nhỏ nhất.");
        Console.WriteLine("3. Dòng có tổng lớn nhất.");
        Console.WriteLine("4. Tổng các số không phải là số nguyên tố.");
        Console.WriteLine("5. Xoá dòng thứ k.");
        Console.WriteLine("6. Xoá cột chứa phần tử lớn nhất.");
        Console.WriteLine("0. Thoát!");
    }

    static void Choose(int[,] matrix, ref int n, ref int m)
    {
        while (true)
        {
            Console.Clear();
            Menu();
            int choice = int.Parse(Console.ReadLine());
            switch (choice)
            {
                case 1:
                    PrintMatrix(matrix);
                    break;
                case 2:
                    FindMaxMin(matrix);
                    break;
                case 3:
                    FindRowWithMaxSum(matrix);
                    break;
                case 4:
                    FindNonPrimeSum(matrix);
                    break;
                case 5:
                    DeleteRow(matrix, ref n);
                    break;
                case 6:
                    DeleteColumn(matrix, ref m);
                    break;
                case 0:
                    return;
                default:
                    Console.WriteLine("Lựa chọn không hợp lệ. Vui lòng nhập lại.");
            }
        }
    }
}
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
int choice = NhapSoNguyen("lựa chọn menu", 0);

switch (choice)
{
    case 0:
        Console.WriteLine("Đã thoát chương trình.");
        return;

    case 1:
        if (IsEmpty(n, m))
        {
            Console.WriteLine("Ma trận hiện đang trống, không
thể xuất!");
            break;
        }
        Console.WriteLine("Ma trận:");
        OutputMatrix(matrix, n, m);
        break;

    case 2:
        if (IsEmpty(n, m))
        {
            Console.WriteLine("Ma trận trống, không thể tìm
phần tử lớn nhất / nhỏ nhất!");
            break;
        }
        OutputMatrix(matrix, n, m);
        Console.WriteLine($"Phần tử lớn nhất:
{Max_Element(matrix, n, m)}");
        Console.WriteLine($"Phần tử nhỏ nhất:
{Min_Element(matrix, n, m)}");
        break;

    case 3:
        if (IsEmpty(n, m))
        {
            Console.WriteLine("Ma trận trống, không thể tìm
dòng có tổng lớn nhất!");
            break;
        }
        OutputMatrix(matrix, n, m);
        Console.WriteLine($"Đòng có tổng lớn nhất là dòng:
{Find_MaxLine(matrix, n, m)}");
        break;

    case 4:
        if (IsEmpty(n, m))
        {
            Console.WriteLine("Ma trận trống, không thể tính
tổng các số không phải số nguyên tố!");
            break;
        }
        OutputMatrix(matrix, n, m);
        Console.WriteLine($"Tổng các số không phải số nguyên
tố: {Sum_NotPrime(matrix, n, m)}");
        break;

    case 5:
        if (IsEmpty(n, m))
        {
            Console.WriteLine("Ma trận trống, không thể xoá
dòng!");
            break;
        }
        OutputMatrix(matrix, n, m);
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
int k = NhapSoNguyen($"chỉ số dòng cần xoá (1->{n})",
1);
Delete_Row(matrix, ref n, m, k);
OutputMatrix(matrix, n, m);
break;

case 6:
if (IsEmpty(n, m))
{
    Console.WriteLine("Ma trận trống, không thể xoá
cột!");
    break;
}
OutputMatrix(matrix, n, m);
Delete_MaxValueColumn(matrix, n, ref m);
OutputMatrix(matrix, n, m);
break;

default:
Console.WriteLine("Lựa chọn không hợp lệ, vui lòng nhập
lại!");
break;
}

Console.ReadKey();
}

// ===== XỬ LÝ MA TRẬN =====

static void ImportMatrix(int[,] matrix, int n, int m)
{
    Random rnd = new Random();
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            matrix[i, j] = rnd.Next(-100, 100);
}

static void OutputMatrix(int[,] matrix, int n, int m)
{
    Console.WriteLine();
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < m; j++)
            Console.Write($"{matrix[i, j],5}");
        Console.WriteLine();
    }
}

static int Max_Element(int[,] matrix, int n, int m)
{
    int max = matrix[0, 0];
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            if (matrix[i, j] > max)
                max = matrix[i, j];
    return max;
}

static int Min_Element(int[,] matrix, int n, int m)
{
    int min = matrix[0, 0];
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            if (matrix[i, j] < min)
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        min = matrix[i, j];
    return min;
}

static int Find_MaxLine(int[,] matrix, int n, int m)
{
    int maxSum = int.MinValue, index = 0;
    for (int i = 0; i < n; i++)
    {
        int sum = 0;
        for (int j = 0; j < m; j++)
            sum += matrix[i, j];

        if (sum > maxSum)
        {
            maxSum = sum;
            index = i;
        }
    }
    return index + 1;
}

static bool IsPrime(int x)
{
    if (x < 2) return false;
    for (int i = 2; i <= Math.Sqrt(x); i++)
        if (x % i == 0) return false;
    return true;
}

static int Sum_NotPrime(int[,] matrix, int n, int m)
{
    int sum = 0;
    for (int i = 0; i < n; i++)
        for (int j = 0; j < m; j++)
            if (!IsPrime(matrix[i, j]))
                sum += matrix[i, j];
    return sum;
}

static void Delete_Row(int[,] matrix, ref int n, int m, int k)
{
    if (k < 1 || k > n)
    {
        Console.WriteLine($"Dòng {k} không tồn tại!");
        return;
    }

    for (int i = k - 1; i < n - 1; i++)
        for (int j = 0; j < m; j++)
            matrix[i, j] = matrix[i + 1, j];

    n--;
    Console.WriteLine($"Đã xoá dòng {k} thành công.");
}

static void Delete_MaxValueColumn(int[,] matrix, int n, ref int m)
{
    int max = Max_Element(matrix, n, m);
    int colIndex = -1;

    for (int i = 0; i < n && colIndex == -1; i++)
        for (int j = 0; j < m; j++)
            if (matrix[i, j] == max)
            {
```

## IT008 – LẬP TRÌNH TRỰC QUAN

```
        colIndex = j;
        break;
    }

    if (colIndex == -1)
    {
        Console.WriteLine("Không tìm thấy phần tử lớn nhất để xoá.");
        return;
    }

    for (int j = colIndex; j < m - 1; j++)
        for (int i = 0; i < n; i++)
            matrix[i, j] = matrix[i, j + 1];

    m--;
    Console.WriteLine($"Đã xoá cột {colIndex + 1} (chứa phần tử lớn
nhất {max});");
}
}
```

Bảng 6.2 - Testcase 1 của bài 6

Input:	Output:
2	Ma trận:
3	
1	-100 85 -43 82 82 59
2	-100 85 -43 82 82 59
0	Phần tử lớn nhất: 85 Phần tử nhỏ nhất: -100 Đã thoát chương trình.

## IT008 – LẬP TRÌNH TRỰC QUAN

```

----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 1
Ma trận:
-100 85 -43
 82 82 59
----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 2
-100 85 -43
 82 82 59
Phần tử lớn nhất: 85
Phần tử nhỏ nhất: -100
----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 0
Đã thoát chương trình.

```

Hình 30. Testcase I của bài 6

Bảng 6.3 - Testcase 2 của bài 6

<b>Input:</b>	<b>Output:</b>
3	Ma trận:
3	
1	-10 -92 17 -86 -46 26 68 9 0
5	-10 -92 17
-5	-86 -46 26
6	68 9 0 chỉ số dòng cần xoá (1->3) phải $\geq 1$ , vui lòng nhập lại! Dòng 6 không tồn tại!
	-10 -92 17 -86 -46 26 68 9 0
6	-10 -92 17 -86 -46 26 68 9 0 Đã xoá cột 1 (chứa phần tử lớn nhất 68).
	-92 17 -46 26 9 0
0	Đã thoát chương trình

## IT008 – LẬP TRÌNH TRỰC QUAN

<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 1 Ma trận:  -10 -92 17 -86 -46 26 68 9 0 </pre>	<pre> 0. Thoát! Nhập lựa chọn menu: 5 -10 -92 17 -86 -46 26 68 9 0 Nhập chỉ số dòng cần xoá (1-&gt;3): -5 chỉ số dòng cần xoá (1-&gt;3) phải ≥ 1, vui lòng nhập lại! Nhập chỉ số dòng cần xoá (1-&gt;3): 6 Đòng 6 không tồn tại!  -10 -92 17 -86 -46 26 68 9 0 </pre>
<pre> Nhập lựa chọn menu: 6 -10 -92 17 -86 -46 26 68 9 0 Đã xoá cột 1 (chứa phần tử lớn nhất 68).  -92 17 -46 26 9 0 </pre>	<p><b>0. Thoát!</b></p> <p>Nhập lựa chọn menu: 0</p> <p>Đã thoát chương trình.</p>

Hình 31. Testcase 2 của bài 6

Bảng 6.4 - Testcase 3 của bài 6

Input:	Output:
3	Ma trận:
3	
1	-10 -35 24 2 41 -28 64 -24 32
2	-10 -35 24 2 41 -28 64 -24 32 Phần tử lớn nhất: 64 Phần tử nhỏ nhất: -35
5	
3	-10 -35 24 2 41 -28 64 -24 32
	Dã xoá dòng 3 thành công.
	-10 -35 24 2 41 -28
2	-10 -35 24 2 41 -28 Phần tử lớn nhất: 41 Phần tử nhỏ nhất: -35
0	Đã thoát chương trình.

## IT008 – LẬP TRÌNH TRỰC QUAN

<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 1 Ma trận: -10 -35 24  2 41 -28  64 -24 32 Nhấn phím bất kỳ để quay lại menu... </pre>	<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 2 -10 -35 24  2 41 -28  64 -24 32 Phần tử lớn nhất: 64 Phần tử nhỏ nhất: -35 Nhấn phím bất kỳ để quay lại menu... </pre>
<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 5 -10 -35 24  2 41 -28  64 -24 32 Nhập chỉ số dòng cần xoá (1-&gt;3): 3 Đã xoá dòng 3 thành công. -10 -35 24  2 41 -28 Nhấn phím bất kỳ để quay lại menu... </pre>	<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 2 -10 -35 24  2 41 -28 Phần tử lớn nhất: 41 Phần tử nhỏ nhất: -35 Nhấn phím bất kỳ để quay lại menu... </pre>
<pre> ----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 0 Đã thoát chương trình. </pre>	

Hình 32. Testcase 3 của bài 6

Bảng 6.5 - Testcase 4 của bài 6

<b>Input:</b>	<b>Output:</b>
-3	số dòng phải ≥ 1, vui lòng nhập lại!
3	số cột phải ≥ 1, vui lòng nhập lại!
-2	số cột phải ≥ 1, vui lòng nhập lại!
0	
2	Ma trận:
1	<pre> 33 5 -19 31 18 -59 </pre>

## IT008 – LẬP TRÌNH TRỰC QUAN

6	<pre> 33 5 -19 31 18 -59  5 31 -59 Đã xoá cột 1 (chứa phần tử lớn nhất 31). </pre>
4	<pre> 5 31 -59 Tổng các số không phải số nguyên tố: -59 </pre>
6	<pre> 5 31 -59 Đã xoá cột 1 (chứa phần tử lớn nhất 31). </pre>
1	Ma trận hiện đang trống, không thể xuất!
2	Ma trận trống, không thể tìm phần tử lớn nhất / nhỏ nhất!
0	Đã thoát chương trình.

```

==== NHẬP KÍCH THƯỚC CỦA MA TRẬN ====
Nhập số dòng: -3
số dòng phải ≥ 1, vui lòng nhập lại!
Nhập số dòng: 3
Nhập số cột: -2
số cột phải ≥ 1, vui lòng nhập lại!
Nhập số cột: 0
số cột phải ≥ 1, vui lòng nhập lại!
Nhập số cột: 2

```

```

----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 1
ma trận:
33 5
-19 31
18 -59

```

```

----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 6
33 5
-19 31
18 -59
Đã xoá cột 1 (chứa phần tử lớn nhất 33).

5
31
-59

```

```

----- MENU -----
1. Xuất ma trận.
2. Phần tử lớn nhất / nhỏ nhất.
3. Dòng có tổng lớn nhất.
4. Tổng các số không phải là số nguyên tố.
5. Xoá dòng thứ k.
6. Xoá cột chứa phần tử lớn nhất.
0. Thoát!
Nhập lựa chọn menu: 4
5
31
-59
Tổng các số không phải số nguyên tố: -59
|
```

## IT008 – LẬP TRÌNH TRỰC QUAN

<pre>----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 6        5      -31     -59 Đã xoá cột 1 (chứa phần tử lớn nhất 31).</pre>	<pre>----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 1 Ma trận hiện đang trống, không thể xuất!</pre>
<pre>----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 2 Ma trận trống, không thể tìm phần tử lớn nhất / nhỏ nhất!  </pre>	<pre>----- MENU ----- 1. Xuất ma trận. 2. Phần tử lớn nhất / nhỏ nhất. 3. Dòng có tổng lớn nhất. 4. Tổng các số không phải là số nguyên tố. 5. Xoá dòng thứ k. 6. Xoá cột chứa phần tử lớn nhất. 0. Thoát! Nhập lựa chọn menu: 0 Đã thoát chương trình.</pre>

Hình 33. Testcase 4 của bài 6

### 7. Link source code GitHub:

<https://github.com/sinhdt-926/IT008.Q14>