



Dữ liệu & Cấu trúc điều khiển

1/ (1 điểm) Viết một chương trình hiển thị ra màn hình các giá trị như sau:

N	10*N	100*N	1000*N
1	10	100	1000
2	20	200	2000
3	30	300	3000
4	40	400	4000
5	50	500	5000

Bài làm:

```
package test;

public class Bai1 {
    public static void main(String[] args) {
        System.out.println("N\t10*N\t100*N\t1000*N");
        for(int i=1;i<=5;i++) {
            System.out.println(i+"\t"+10*i+"\t"+100*i+"\t"+1000*i);
        }
    }
}
```

2/ (1 điểm) Viết một chương trình nhập vào số đo chiều dài cạnh của một hình vuông. In ra màn hình một hình vuông rỗng được tạo nên từ các kí tự '*' với số đo chiều dài cạnh đó.

Ví dụ:

```
Enter length of side:15
*****
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*               *
*****
```

Bài làm:

```
package test;
import java.util.Scanner;

public class Bai2 {
    public static void main(String[] args) {
        System.out.println("Nhập chiều dài cạnh:");
        Scanner sc = new Scanner(System.in);
        int m = sc.nextInt();

        // Cách 1:
        for (int i = 0 ; i < m; i++) { // i --> dòng
            for (int j = 0; j < m; j++) { // j --> cột
                if (i==0 || i==m-1 || j==0 || j==m-1) System.out.print(" * ");
                else System.out.print("  ");
            }
            System.out.println();
        }

        // Cách 2:
        for(int i=1; i<=m; i++) {

            if(i==1 || i==m) {
                for(int j=1; j<=m; j++) {
                    System.out.print("*");
                }
            }
            else {
                for(int k=1; k<=m; k++) {
                    if(k==1 || k==m) System.out.print("*");

                    else System.out.print(" ");
                }
            }

            System.out.println();
        }
    }
}
```

3/ (2 điểm) Một số được gọi là đối xứng nếu khi đọc từ trái qua phải hay đọc từ phải qua trái của số đó thì ta đều thu được một con số giống nhau. Ví dụ các số 12321, 55555, 45554 và 11611 đều là những con số đối xứng. Hãy viết chương trình nhập vào một số nguyên có 7 chữ số và kiểm tra xem số đó có phải là một số đối xứng hay không

Bài làm:

```
package test;
import java.util.Scanner;
public class Bai3 {
    public static void main(String[] args) {
        // Cách 1:
        // Thuật toán đảo ngược số
        Scanner sc = new Scanner(System.in);
```

```
int n = sc.nextInt(), k = n, m, kq = 0;

while (k != 0) {
    m = k % 10;
    kq = kq * 10 + m;
    k = k / 10;
}

if(kq == n) System.out.println("Số đối xứng");
else System.out.println("Số không đối xứng");

// Cách 2:
// Dùng cấu trúc dữ liệu mảng
int kt = 1;
System.out.println("nhập số phần tử trong mảng: ");
n = sc.nextInt();

int A[] = new int[n];

System.out.println("nhập các giá trị phần tử: ");
for (int i = 0; i < n; i++) {
    System.out.print("A[" + i + "] = ");
    A[i] = sc.nextInt();
}

for (int i = 0; i < n / 2; i++) {
    if (A[i] != A[n - i - 1]) {
        kt = 0;
        break;
    }
}

if (kt == 0) System.out.println("không phải là số đối xứng");
else System.out.println("là số đối xứng");
}
```

4/ (2 điểm) Một ngân hàng muốn thiết lập một hệ thống truyền tin qua đường truyền Internet nhưng lo ngại dữ liệu truyền đi của khách hàng có thể bị rò rỉ (xem trộm). Bạn hãy viết một chương trình mã hóa dữ liệu của ngân hàng trước khi truyền đi để nâng cao tính bảo mật của khách hàng. Biết rằng mỗi dữ liệu truyền được nhập vào chương trình của bạn là một số nguyên không âm có 4 chữ số. Yêu cầu: chương trình của bạn sẽ đọc số nguyên có 4 chữ số đó do người dùng nhập vào và mã hóa theo các qui tắc tuần tự như sau:

a/ Thay mỗi chữ số trong số nguyên trên bằng một chữ số mới được tạo ra bằng cách cộng thêm 7 và lấy phần dư khi chia cho 10

b/ Tráo đổi giá trị của số thứ 1 với số thứ 3 và số thứ 2 với số thứ 4

Sau cùng, hãy in dữ liệu đã được mã hóa ra màn hình

Ví dụ: dữ liệu nhập vào 3478 → dữ liệu mã hóa 4501

Bài làm:

```
package lab_assignment2;

import java.util.Scanner;

public class Bai4 {
    public static void main(String[] args) {
        System.out.print("Nhập vào một gói dữ liệu của khách hàng: ");
        Scanner input;
        int data;

        do {
            input = new Scanner(System.in);
            data = input.nextInt();
        } while (data < 0);

        String[] maHoa = new String[4];
        int viTri = maHoa.length - 1;

        while (data != 0) {
            int donVi = data % 10;
            donVi = (donVi + 7) % 10;
            maHoa[viTri] = donVi + "";
            data = data / 10;
            viTri--;
        }

        String temp;
        temp = maHoa[0]; maHoa[0] = maHoa[2]; maHoa[2] = temp; // Tráo trị
        temp = maHoa[1]; maHoa[1] = maHoa[3]; maHoa[3] = temp; // Tráo trị

        System.out.print("Dữ liệu đã được mã hóa: ");

        for (int i=0; i<maHoa.length; i++) {
            System.out.print(maHoa[i]);
        }
    }
}
```

5/ (4 điểm) Giai thừa của một số nguyên không âm n được viết là $n!$ (“ n giai thừa”) và được tạo ra bằng cách như sau:

$$n! = n * (n - 1) * (n - 2) * \dots * 1 \quad (n \geq 1)$$

và $n! = 1$ (nếu $n = 0$)

a/ Viết chương trình nhập vào một giá trị n và tính $n!$

b/ Viết chương trình tính giá trị xấp xỉ của e (hằng số toán học) bằng công thức sau:

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}$$

(với n càng lớn thì e có độ chính xác càng cao)

c/ Viết chương trình tính giá trị của e^x có được từ công thức sau:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

(với n càng lớn thì e^x có độ chính xác càng cao)

```
package lab_assignment2;
import java.util.Scanner;

public class Bai5 {
    static int GiaiThua(int n) {
        int giaiThua = 1;
        for (int i=1; i<=n; i++) {
            giaiThua = giaiThua * i;
        }
        return giaiThua;
    }

    public static void main(String[] args) {
        Scanner input;
        int n = 0;

        do {
            System.out.print("Nhập vào một số nguyên không âm: ");
            input = new Scanner(System.in);
            n = input.nextInt();
        } while(n < 0);

        System.out.println("Câu a: " + n + "! = " + GiaiThua(n));

        double e = 0;
        for (int i=0; i<=n; i++) {
            e = e + 1.0 / GiaiThua(i);
        }

        System.out.println("Câu b: e = " + e);

        System.out.print("Nhập vào một số nguyên không âm x: ");
        int x = input.nextInt();
        double ex = 0;
        for (int i=0; i<=n; i++) {
            ex = ex + Math.pow(x, i) / GiaiThua(i);
        }

        System.out.println("Câu c: e^x = " + ex);
    }
}
```