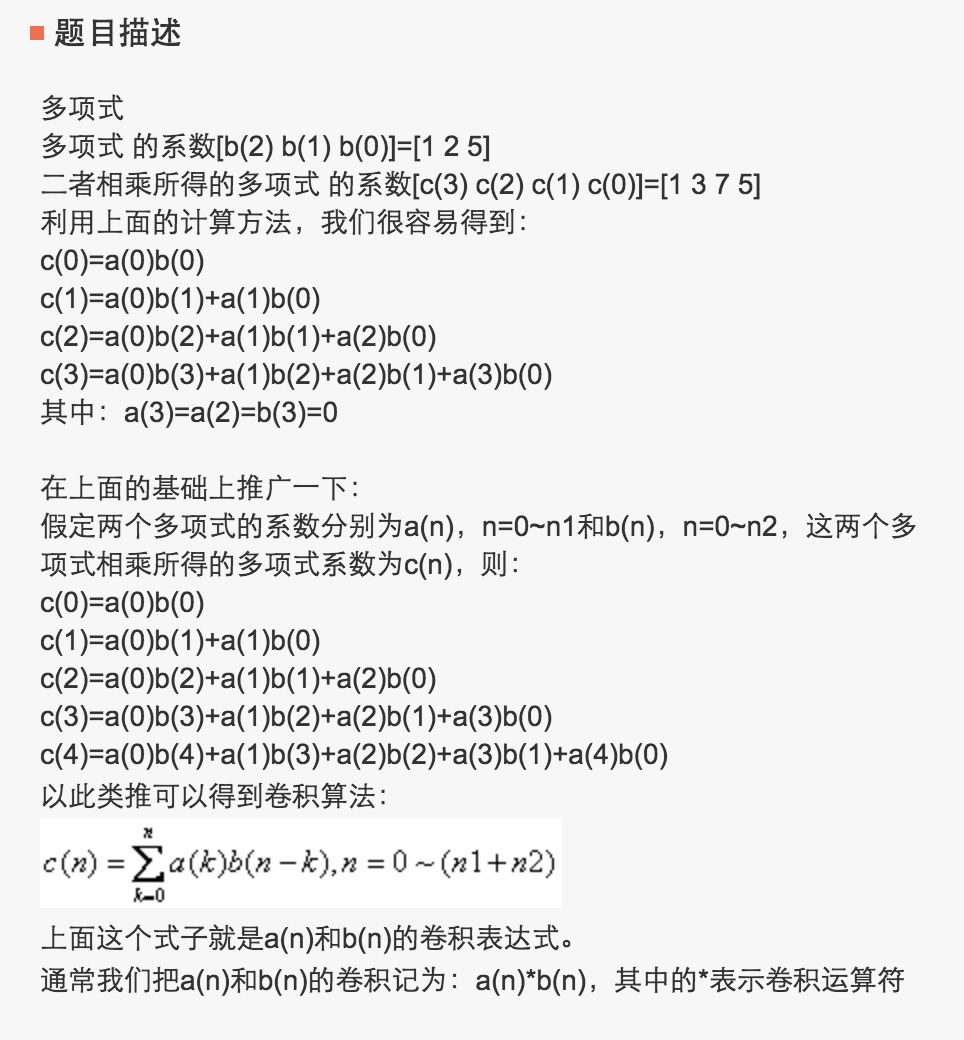
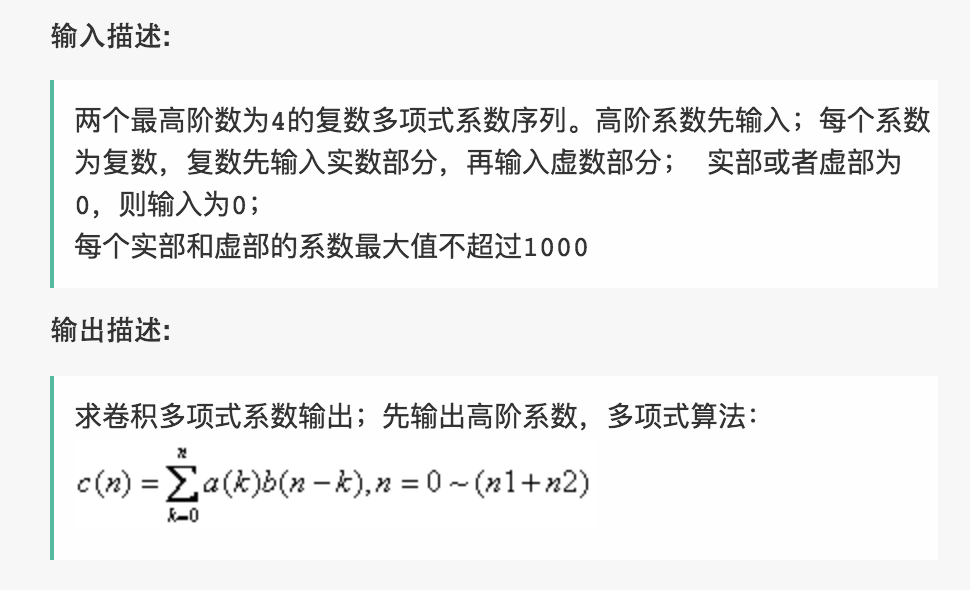
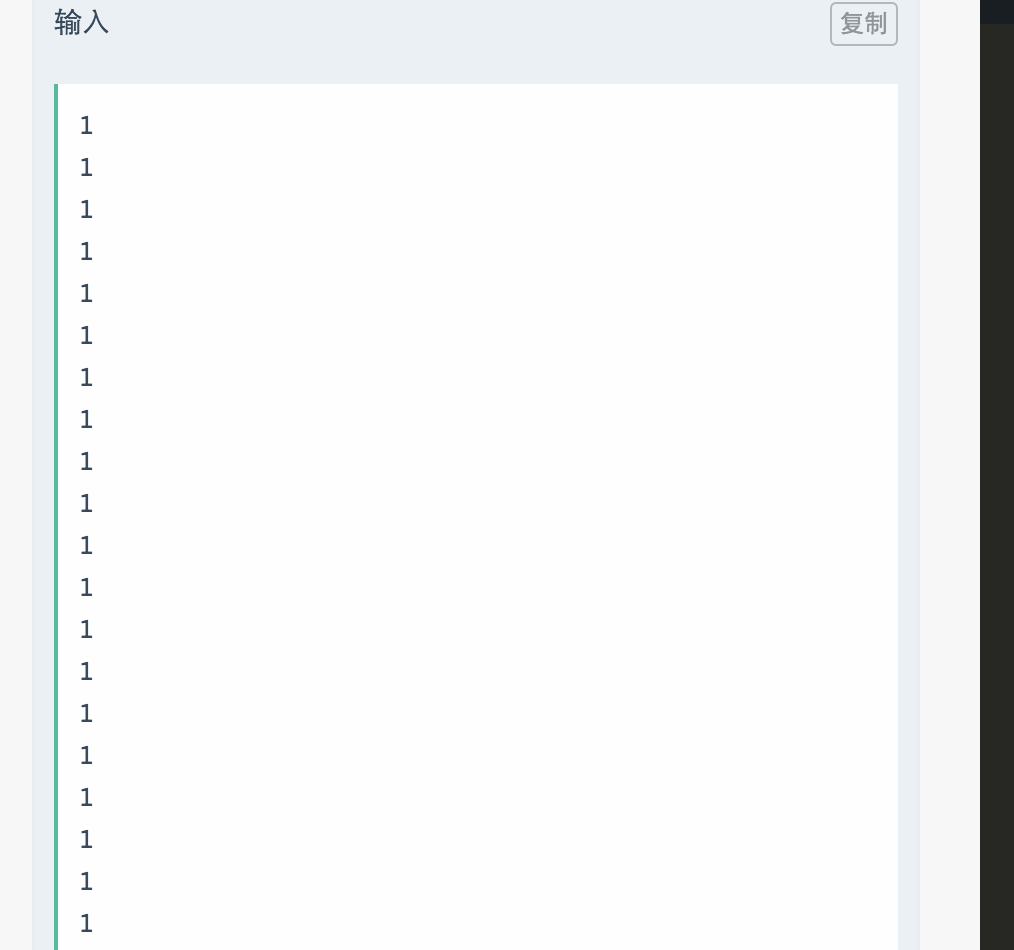
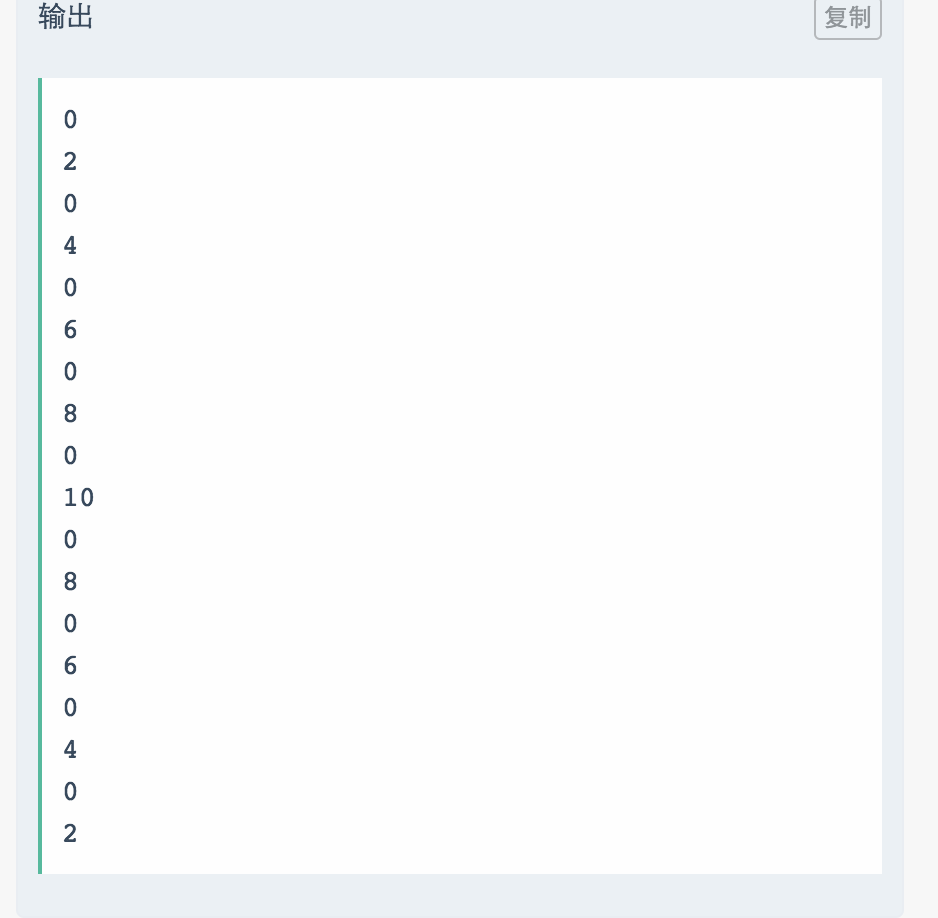


|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | import java.util.HashMap;  import java.util.Iterator;  import java.util.Scanner;    public class Main {        public static void main(String[] args) {          Scanner in = new Scanner(System.in);          String str;          str = in.nextLine();          int i, ans = 0, tmp = 0, flag = 1;//1表示正整数          for (i = 0; i < str.length(); i++) {              if (str.charAt(i) >= '0' && str.charAt(i) <= '9') {                  tmp = tmp \* 10 + flag \* (str.charAt(i) - '0');                  if(i<str.length()&&(str.charAt(i+1)<'0'||str.charAt(i+1)>'9')){                      ans+=tmp;                      tmp=0;                      flag=1;                  }              } else if (str.charAt(i) == '-') {                  flag = flag == 1 ? -1 : 1;              } else {                  ans += tmp;                  tmp = 0;                  flag = 1;              }          }          System.out.println(ans + tmp);      }  } |

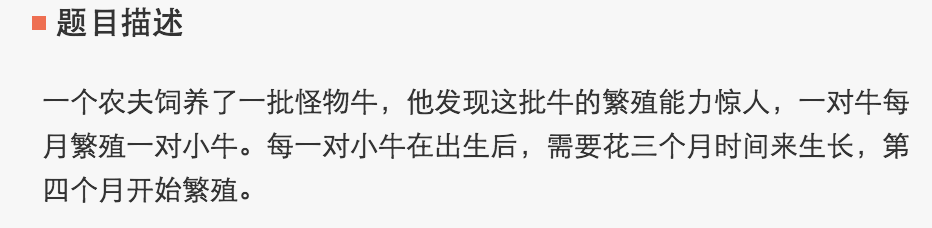


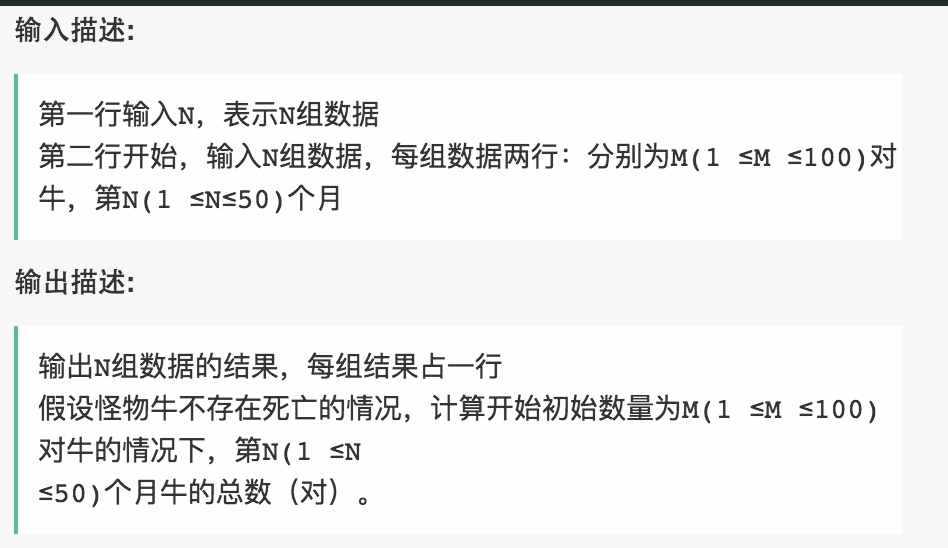




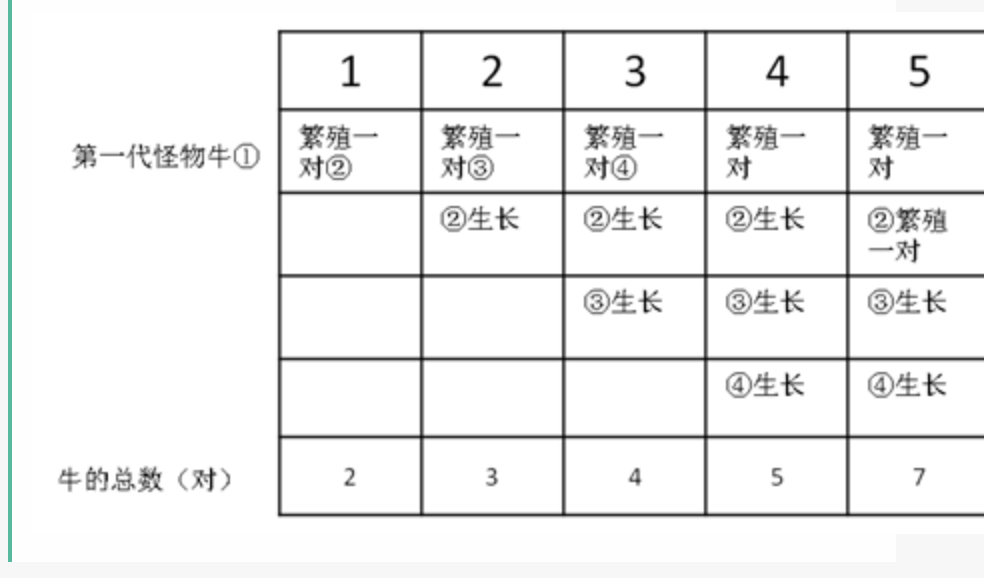


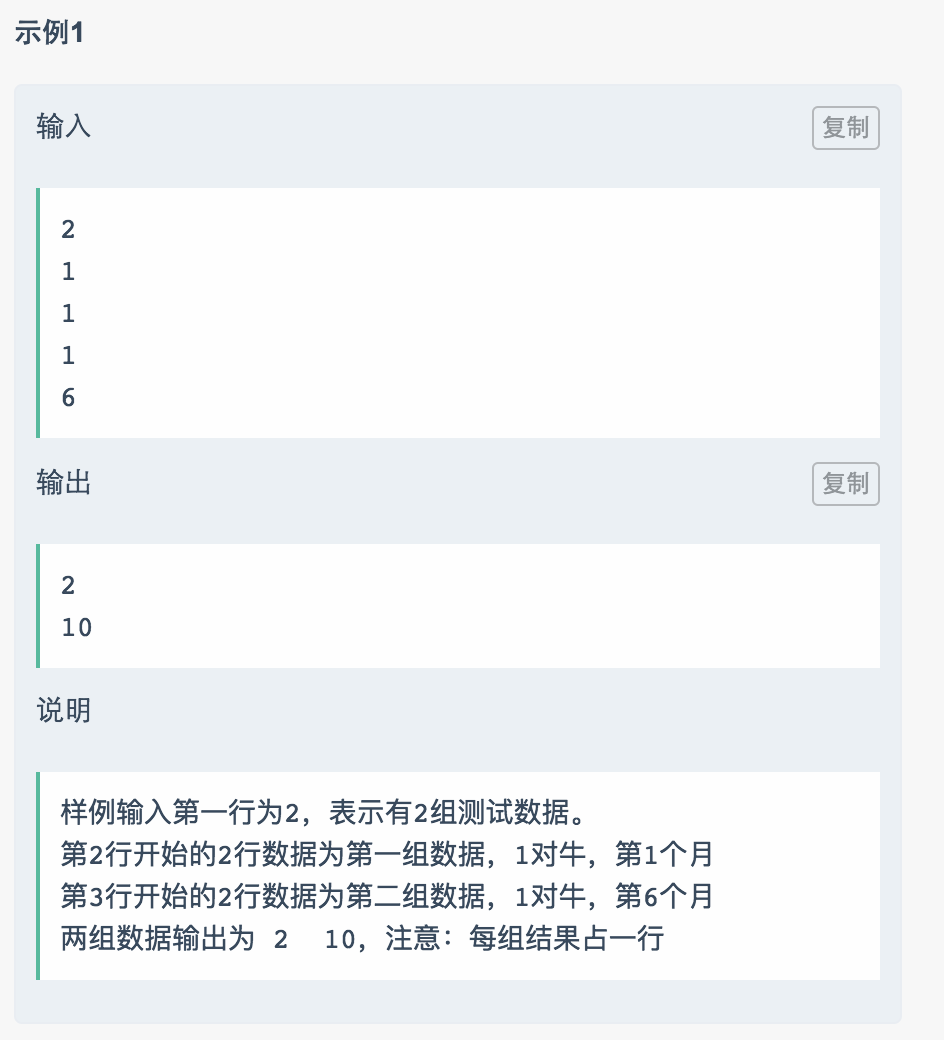
|  |
| --- |
| import java.util.HashMap;  import java.util.Iterator;  import java.util.Scanner;    public class Main {        public static void main(String[] args) {          Scanner in = new Scanner(System.in);          int t, n, m, t1, i;          int[] f = new int[60];          t = in.nextInt();          for (t1 = 0; t1 < t; t1++) {              n = in.nextInt();              m = in.nextInt();              for (i = 1; i <= 4; i++) {                  f[i] = n \* (i+1);              }              for (i = 5; i <= m; i++)                  f[i] = f[i - 1] + f[i - 4];              System.out.println(f[m]);          }      }  } |











public class Main {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int[] a = new int[10];

        int[] b = new int[10];

        for (int i = 0; i<10; i++) {

            a[i] = sc.nextInt();

        }

        for (int i = 0; i<10; i++) {

            b[i] = sc.nextInt();

        }

        //(a + bi) \* (c + di) = (a\*c - b\*d) + (a\*d + b\*c)i

        int[] c = new int[18];

        c[0] = getReal(a[0], a[1], b[0], b[1]);

        c[1] = getXu(a[0], a[1], b[0], b[1]);

        c[2] = getReal(a[2],a[3],b[0], b[1]) + getReal(a[0],a[1],b[2],b[3]);

        c[3] = getXu(a[2],a[3],b[0], b[1]) + getXu(a[0],a[1],b[2],b[3]);

        c[4] = getReal(a[4],a[5],b[0], b[1]) + getReal(a[2],a[3],b[2],b[3]) + getReal(a[0],a[1],b[4],b[5]);

        c[5] = getXu(a[4],a[5],b[0], b[1]) + getXu(a[2],a[3],b[2],b[3]) + getXu(a[0],a[1],b[4],b[5]);

        c[6] = getReal(a[6], a[7],b[0], b[1])+getReal(a[4],a[5],b[2],b[3])+getReal(a[2],a[3],b[4],b[5])+getReal(a[0],a[1],b[6],b[7]);

        c[7] = getXu(a[6], a[7],b[0], b[1])+getXu(a[4],a[5],b[2],b[3])+getXu(a[2],a[3],b[4],b[5])+getXu(a[0],a[1],b[6],b[7]);

        c[8] = getReal(a[8], a[9],b[0], b[1])+getReal(a[6], a[7],b[2], b[3]) +getReal(a[4],a[5],b[4],b[5])

                + getReal(a[2],a[3],b[6],b[7]) + getReal(a[0],a[1],b[8],b[9]);

        c[9] = getXu(a[8], a[9],b[0], b[1])+getXu(a[6], a[7],b[2], b[3]) +getXu(a[4],a[5],b[4],b[5])

                + getXu(a[2],a[3],b[6],b[7]) + getXu(a[0],a[1],b[8],b[9]);

        c[10] = getReal(a[2],a[3],b[8],b[9]) + getReal(a[4],a[5],b[6],b[7])+getReal(a[6],a[7],b[4],b[5])+

                + getReal(a[8],a[9],b[2],b[3]);

        c[11] = getXu(a[2],a[3],b[8],b[9]) + getXu(a[4],a[5],b[6],b[7])+getXu(a[6],a[7],b[4],b[5])+

                + getXu(a[8],a[9],b[2],b[3]);

        c[12] = getReal(a[4],a[5],b[8],b[9]) + getReal(a[6],a[7],b[6],b[7]) +getReal(a[8],a[9],b[4],b[5]);

        c[13] = getXu(a[4],a[5],b[8],b[9]) + getXu(a[6],a[7],b[6],b[7]) +getXu(a[8],a[9],b[4],b[5]);

        c[14] = getReal(a[6],a[7],b[8],b[9]) + getReal(a[8],a[9],b[6],b[7]);

        c[15] = getXu(a[6],a[7],b[8],b[9]) + getXu(a[8],a[9],b[6],b[7]);

        c[16] = getReal(a[8],a[9],b[8],b[9]);

        c[17] = getXu(a[8],a[9],b[8],b[9]);

        for (int num : c) {

            System.out.println(num);

        }

    }

    private static int getReal(int a, int b, int c, int d) {

        return a\*c - b\*d;

    }

   private static int getXu(int a, int b, int c, int d) {

        return a\*d + b\*c;

    }

}