Question 1

Iterator

Given a number N followed by N integers, it will comprise of certain positive numbers, then a zero then follows negative numbers. Find the minimum number of iterations to convert a Positive-Zero-Negative array into a Negative-Zero-Positive array, by abiding by the given constraints.
Constraints for Swapping:
1. Swapping is possible only with a Zero
2. A positive number should be swapped only to the right hand side
3. A negative number should be swapped only to the left hand side
4. Swapping can be done to a maximum of two steps at a time.
Find the minimum number of iterations.

Input Format:

The first line of input has a single integer value N followed by N integer values.

Input Constraints:

3<=N<=2500 -1000<=a[i]<=1000 Where N is the number of elements of the array and a[i] is each element of the array.

Output Format:

One single integer denoting the number of iterations required.

Sample Input:

348
236 167 776 769 752 381 648 23 968 687 837 733 656 481 659 608 913 25 750 474 876 574 410 942 390 230 869 332 523 658 536 243 189 412 984 318 986 532 321 588 376 883 788 258 434 896 838 201 129 363 626 628 793 94 768 710 53 169 106 679 470 356 537 72 978 647 182 185 158 830 121 946 785 87 979 657 953 358 478 588 280 300 895 596 3 952 184 325 712 711 237 474 815 541 714 166 50 730 569 309 418 857 624 680 480 838 443 153 900 702 492 629 520 196 871 170 224 436 430 267 345 433 850 394 132 4 60 192 262 547 805 104 296 4 589 557 340 653 316 682 973 186 855 980 659 422 647 284 265 781 296 410 539 318 921 857 138 531 75 14 140 443 87 5 73 786 686 0 -2 -452 -224 -857 -700 -854 -536 -710 -541 -873 -962 -365 -385 -175 -937 -845 -722 -108 -893 -782 -462 -346 -140 -728 -971 -308 -400 -421 -626 -434 -775 -161 -501 -974 -605 -893 -171 -848 -767 -146 -788 -523 -788 -392 -962 -290 -828 -314 -701 -45 -317 -323 -956 -805 -14 -416 -326 -982 -99 -168 -959 -450 -862 -544 -274 -401 -412 -380 -917 -991 -533 -759 -967 -1000 -859 -675 -130 -663 -251 -759 -73 -632 -405 -776 -193 -980 -690 -433 -369 -518 -666 -758 -884 -127 -9 -355 -507 -853 -840 -304 -208 -584 -562 -527 -175 -566 -379 -503 -749 -524 -596 -761 -860 -111 -950 -834 -16 -407 -79 -420 -586 -830 -58 -523 -979 -646 -364 -496 -700 -283 -543 -241 -219 -327 -169 -262 -252 -231 -330 -314 -650 -331 -731 -523 -502 -185 -395 -458 -181 -222 -708 -47 -999 -941 -507 -896 -773 -952 -516 -240 -53 -783 -696 -326 -541 -170 -280 -556 -889 -155 -510 -145 -233 -291 -663 -532 -31 -270 -521 -672 -863

Sample Output:

30393

CODE:

```
import java.util.*;
class Main{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        int n=sc.nextInt();
        int a[]=new int[n];
        int pos=0,neg=0;
        for(int i=0;i<n;i++)
        {
            a[i]=sc.nextInt();
            if(a[i]>0)
            {
                pos++;
            }
            else if(a[i]<0)
            {
                neg++;
            }
        }
        System.out.print(pos*neg+(pos+neg));

    }
```

```
}
```

Question 2

Greedy Thief

Victor is a businessman, he usually carries his money in a briefcase. This briefcase is highly secure and it is protected with a digital number lock. The briefcase is made of proto-Adamantium, an alloyed mixture of Adamantium and Vibranium. Even a blast will not be able to break it. Only Thanos has the power of breaking it.

Even though it is so secure, if anyone enters the right number combination in the number lock, it will open up. So for safety purpose, he changes the number lock everyday. Everyday, he carries a different amount in the briefcase, and he sets the lock with a minimum possible number whose sum of digits will be equal to the money present in the briefcase.

Loki, assistant of victor, wants to Steele the money and run away. He somehow came to know the money which Victor had inside the briefcase. Help Loki to identify the number lock, so he can take the money.

Input Format:

One single integer N

Input Constraints:

1<=N<=10000000000000000000000000

Output Format:

A smallest possible number whose sum of digits equals to N

Sample Input:

11

Sample Output:

29

CODE

```java
import java.util.*;
class Main{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        long n=sc.nextLong();
        long d=n%9;
        long m=n/9;
        if(d!=0)
        {
            System.out.print(d);
        }

        for(long i=1;i<=m;i++)
        {
            System.out.print("9");
        }
    }
}
```

Question 3

Chocolate Truffle

Willy Wonka has N chocolate truffles with him, he wants to distribute these to N different people such that any person gets either 0, 1 or 2 truffle(s),But the first person should have any number of truffle(s). Help Willy Wonka to evaluate the number of ways he can do this.

For example, N=2, the truffle can be distributed as {(1,1),(2,0),(0,2)}. But we will consider only the first two cases as the third case just another representation of case two.

For example, N=3, the truffle can be distributed as {(1,1,1),(2,1,0),(1,2,0),(1,0,2), (0,1,2), (2,0,1),(0,2,1)}. But we will consider only the first three cases as the other cases are just another representation of the first three cases (the order of occurance

Input Format:

Single line, containing a single integer N

Input Constraints:

1 < N < 50, where N is the number of chocolate truffles with Willy Wonka

Output Format:

Single line, with single integer denoting the number of ways Willy Wonka can do this.

Sample Input:

19

Sample Output:

6765

CODE:

```
import java.util.*;
class Main{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        int n=sc.nextInt();
        int a=0,b=1,c=0;
        for(int i=0;i<n;i++)
        {
            c=a+b;
            a=b;
            b=c;
        }
        System.out.print(c);
    }
}
```

Question 4

The Creator - Hollow Square

Lying of the roof top, at night time, enjoying thee cold breeze, seeing the stars is a enjoyable moment. It is believed that if we think of a particular shape in in mind, and look at the stars, it will seem like the stars are aligned in that particular pattern.

Emma is a 5 year old kid, she is looking at the stars and expecting the stars to come in a particular pattern. Victor has a super power to make the stars come in a particular alignment. Can you help Victor to make Emma happy.

Emma thinks of a hollow square now.

Input Format:

One single integer input N, where N is the size of the grid

Input Constraints:

1<=N<=50

Output Format:

The shape thought by Emma

Sample Input:

5

Sample Output:

```
*****
* *
* *
* *
*****
```

CODE:

```java
import java.util.*;
class Main{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        int n=sc.nextInt();
        for(int i=0;i<n;i++)
        {
            for(int j=0;j<n;j++)
            {

            if (i > 0 && i < n - 1 && j > 0
                && j < n- 1) {
                System.out.printf(" ");
            }
            else {
                System.out.printf("*");
            }

            }
            System.out.print("\n");
        }
    }
}
```

Question 6

Dictionary

We all know some of the greatest dictionaries in the world like the Oxford English Dictionary or the Cambridge English Dictionary so on and so forth. Ram, an English enthusiast who also happens to be a very good programmer is very interested in building a digital dictionary. He would like to add a feature to the digital dictionary that he believes is "cool". Here is an excerpt that describes this feature.
Given any sequence of characters between 'a' and 'z' (both inclusive), the tool needs to return the row in which the word will possibly occur if all the possible arrangements of the given sequence is taken and arranged lexicographically.
Since Ram is busy building the other features of the digital dictionary, you have been given the job of building this "cool" feature.
INPUT FORMAT
The only line of input contains the string S.

OUTPUT FORMAT
Print a single integer denoting the Rank of the Word.

INPUT CONSTRAINTS
$1 \leq |S| \leq 20$
$'a' \leq S[i] \leq 'z'$

Input Format:

The only line of input contains the string S.

Input Constraints:

$1 \leq |S| \leq 20$

$'a' \leq S[i] \leq 'z'$

Output Format:

Print a single integer denoting the Rank of the Word.

Sample Input:

word

Sample Output:

22

CODE:

```
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        String s=sc.nextLine();
        int len=s.length();
        int d=len;
        int a[]=new int[len];
        int b[]=new int[len];
        for(int i=0;i<len;i++)
        {
            a[i]=(int)s.charAt(i);
            b[i]=(int)s.charAt(i);

        }
        Arrays.sort(b);
        /*for(int i=0;i<len;i++)
        {
            System.out.print(a[i]+" ");
        }
        for(int i=0;i<len;i++)
        {
            System.out.println(b[i]+" ");
```

```
        }*/
        for(int i=0;i<len;i++)
        {
            int x1=a[i];
            for(int j=0;j<len;j++)
            {
                if(a[i]==b[j])
                {
                    a[i]=j+1;
                }
            }
        }
       /*for(int i=0;i<len;i++)
        {
            System.out.println(a[i]+" ");
        }*/



        for(int i=0;i<len;i++)
        {
            int count=0;
            for(int j=i+1;j<len;j++)
            {
                if(a[i]>a[j])
                {
                    count++;
                }
            }
            b[i]=count;
        }
        /*for(int i=0;i<len;i++)
        {
            System.out.print(b[i]+" ");
        }*/

        int fact=1;
        int o=0;
        int sum=0,index=0;
        for(int i=len-1;i>=0;i--)
        {
            int d1=i;
            for(int j=1;j<=d1;j++)
            {
                fact=fact*j;
            }
            int div=check(o++,a,len);
            sum+=(int)(fact*b[index])/div;
            index++;
            //System.out.println(fact);
            fact=1;
```

```
        }

        System.out.print(sum+1);


    }

     public static int check(int i,int a[],int len)
    {
         int div=1;
        for(int x=i;x<len;x++)
        {
                int c=1;
                for(int y=x+1;y<len;y++)
                {
                    if(a[x]==a[y])
                    {
                        c++;
                    }
                }
                for(int z=1;z<=c;z++)
                {
                    div=div*z;
                }



        }
        //System.out.print(div+" ");
        return div;
    }
}
```

Question 7

Stock Exchange Specialist

Victor is a Stock Exchange Specialist. Any stocks that he buys, he has a capacity of for such greater price. He has created a algorithm to find the profit that can be obtained by trading certain stocks. The machines will be generating a string, the string will be made up of integers from 0 to 9. Now each integer in this string will be a coded value of the profit. You have to multiply a digit N, with n-1,n-2,...till n-i>=1. Each digit in the string will be giving a profit of the particular amount thus obtained. Now summing up all the profits, it should be equal to the entire string considered as a integer input.
If they are equal, then the prediction was right, so print Right, if else print Wrong.

Input Format:

One single string input N, where N is the prediction by Victor

Input Constraints:

1<=length of N<=10^5

Output Format:

1<=length of N<=10^5

Sample Input:

40583

Sample Output:

Wrong

CODE:

```
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        int n=sc.nextInt();
        int d=n;
        int fact=1,sum=0;
        while(d!=0)
        {
            int d1=d%10;
            for(int i=1;i<=d1;i++)
            {
                fact=fact*i;
            }
            //System.out.print(fact+" ");
            sum=sum+fact;
            fact=1;
            d=d/10;
        }
        if(sum==n)
        {
            System.out.print("Right");
        }
        else
        {
            System.out.print("Wrong");
        }
    }
}
```

Question 9

Perfect Photographer

Victor is a professional photographer, he has shot several world class photographs. One fine evening, he was walking in the sidewalk of a park. He came across Annie Leibovitz, who is known to be one of the best photographers in the world and she was one of Victor's role model. He quickly rushed to Annie and asked her permission to be her assistant. She gave victor a task. There was a twin before them, Chin and Chan. They both had an identical square kite of equal dimension. They held the kite up-straight and its base was parallel to the ground. You saw that from the distance and you could not differentiate between the two kites, since from your line of sight, both the kites were overlapping each other. The begun to rise the kite, and both the kites went up along the x=y line. But both the kites went up at different speeds. As the square kites are moving at different speeds, now they could visibly see both the kites. Since they are moving up along a same line, the intersection formed visibly between them was also a square. Annie told that she will say the area of the intersection X, and Victor had to click the photo exactly when the overlapping area is equal to X. Help victor with a time at which he has to click the photograph so that he will be able to click the photo at that exact second and get to be the assistant of his role model.

Input Format:

The first line of input has one single integer input N, where N corresponds to the side of the Square kite
The second line of input has two integer inputs S1 and S2, where S1 and S2 are the speeds at which kite 1 and kite 2 rise correspondingly
The last line of input has one single integer input K, where K is the area of intersection between two kites

Input Constraints:

1<=N,S1,S2,K<=10^9

Output Format:

Output specifying the time at which the photo has to be clicked. Round off the value to 2 decimal points.

Sample Input:

```
395
307 232
15
```

Sample Output:

```
7.38
```

CODE:

```java
import java.util.*;
class Main
{
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        int s=sc.nextInt();
        int speed1=sc.nextInt();
        int speed2=sc.nextInt();
        int intersection=sc.nextInt();
        int ans1=Math.abs(speed1-speed2);
        double ans2=s*(Math.sqrt(2));
        double ans3=(Math.sqrt(intersection))*(Math.sqrt(2));
        double ans4=ans2-ans3;
        double answer=ans4/ans1;
        System.out.printf("%.2f",answer);
    }
}
```