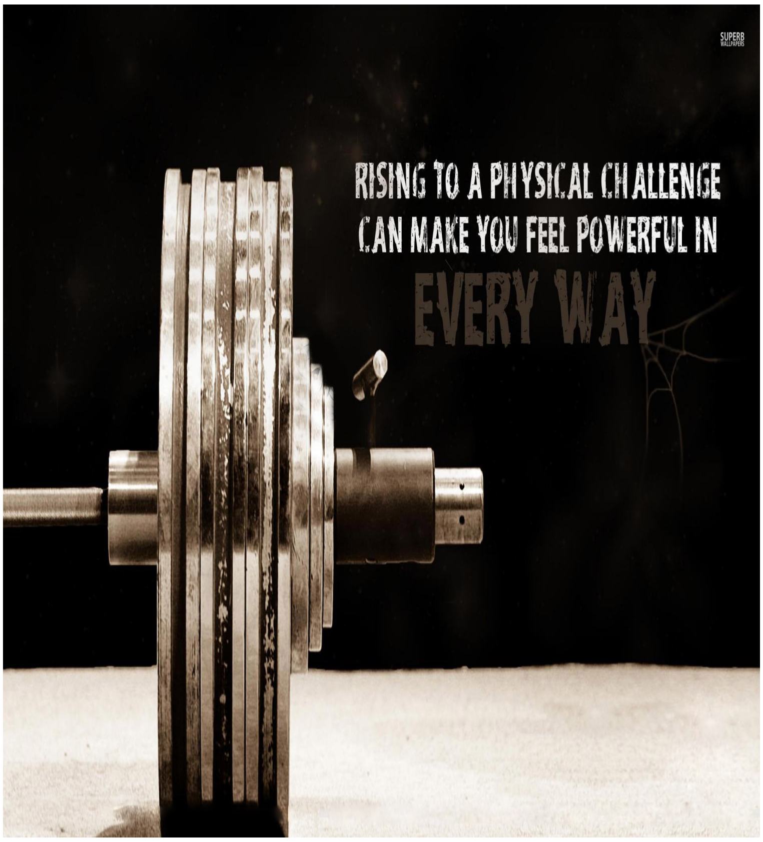
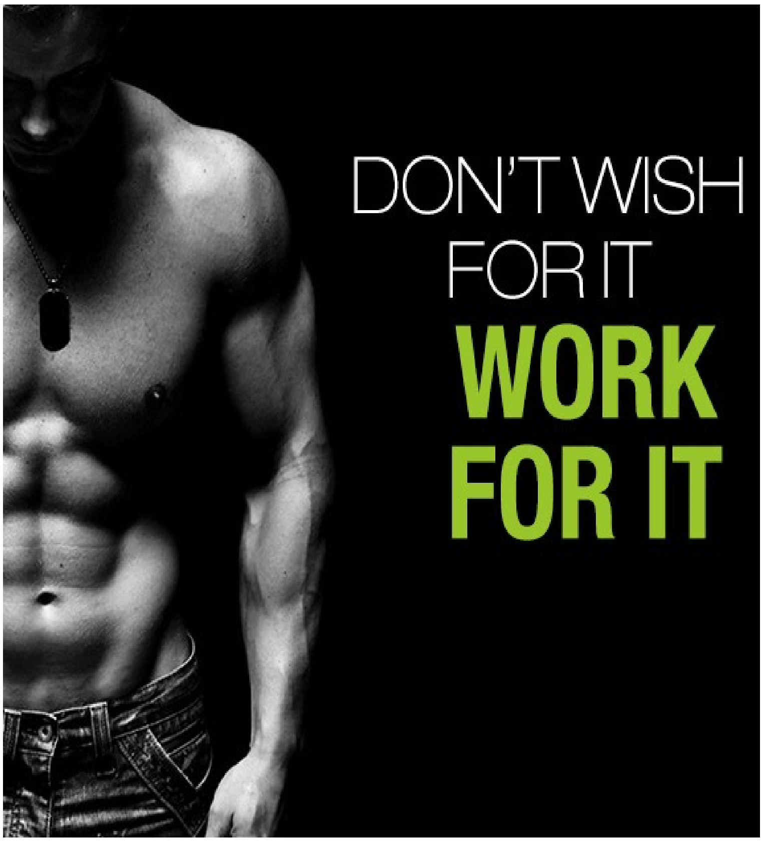
OUTPUT

```
Enter the value of N:
10
Enter the base and exponent of 10 rows
6 13
5 3
43 2
3 8
7 23
5 3
9 92
5 3
11 26
4 23
base  exponent
6      13
5      3
43     2
3      8
7      23
5      3
9      92
5      3
11     26
4      23
Checking function 1: Compare ecodes for 5 index pairs
Enter indices of row1 and row2 that you want to compare:
3 4
ecode of index 1 is greater than ecode of index2
Enter indices of row1 and row2 that you want to compare:
5 6
ecode of index 1 is lesser than ecode of index2
Enter indices of row1 and row2 that you want to compare:
7 1
ecode of index 1 is equal to ecode of index2
Enter indices of row1 and row2 that you want to compare:
2 9
ecode of index 1 is lesser than ecode of index2
Enter indices of row1 and row2 that you want to compare:
3 6
ecode of index 1 is greater than ecode of index2
Checking Function 2: Printing Sorted array
base  exponent
6      13
5      3
43     2
3      8
5      23
7      23
9      92
5      3
11     26
4      23
i_index:3
j_index:7
Checking Function 3: Printing Merged Array
base  exponent
6      13
5      3
5      3
43     2
11     26
3      8
4      23
5      23
7      23
9      92
```

Q. Sorting and Searching- Ecodes

```
import java.util.Scanner;
import java.lang.Math;

public class AB24573 {
     public static int i_index=0,j_index=0;
     public static double findEcode(int audience[][],int
                                                      index){

          double ecode=0;
          ecode=Math.pow(audience[index][0],
                         audience[index][1])%100;
          return ecode;
     }

     public static int comparator(int audience[][],int
                          index1,int index2){
          double ecode1=0,ecode2=0;
          ecode1=findEcode(audience,index1);
          ecode2=findEcode(audience,index2);
          if(ecode1==ecode2)
               return 0;
          else if(ecode1>ecode2)
               return -1;
          else
               return 1;
     }

     public static void sorting(int audience[][],int n){
          int i,j,b,e;

          while(comparator(audience,i_index,i_index+1)==1||
               comparator(audience,i_index,i_index+1)==0){
               i_index++;
          }
          j_index=n-1;
          while(comparator(audience,j_index-1,j_index)==1||
               comparator(audience, j_index,j_index-1)==0){
               j_index--;
          }

          for(i=i_index+1;i<j_index-1;i++){
               for(j=i+1;j<=j_index-1;j++){
                    if(comparator(audience,i,j)==-1){
                         b=audience[i][0];
                         e=audience[i][1];
                         audience[i][0]=audience[j][0];
                         audience[i][1]=audience[i][1];
                         audience[j][0]=b;
                         audience[j][1]=e;
```


SUPERB WALLPAPERS
RISING TO A PHYSICAL CHALLENGE
CAN MAKE YOU FEEL POWERFUL IN
EVERY WAY


DON'T WISH
FOR IT
WORK
FOR IT