

11. (b) Test 11

Test Summary

- No. of Sections: 2
- No. of Questions: 3
- Total Duration: 45 min

Section 1 - Coding Proficiency

Section Summary

- No. of Questions: 2
- Duration: 30 min

Additional Instructions:

None

Q1. **Armstrong Numbers Below the Intervals**
Write a C Program to Display Armstrong Number Between Two Intervals
If there is no elements print No elements

Input Format

Input contains the start and end range

Output Format

Print the values separated by space

Constraints

1 <= start < end <= 1000000

Sample Input

100 500

Sample Output

153 370 371 407

Sample Input

10 100

Sample Output

No elements

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q2. **Odd Position Sorting**
Given 5,1,4,7,9....do alternate sort (odd position sorting) for this..and print 4,5,9

Input Format

Input contains the array size and the values

Output Format

Print the sorted alternate elements

Constraints

1<=size<=1000

Sample Input

10
7 3 1 8 4 9 5 6 2 10

Sample Output

1 2 4 5 7

Sample Input

Sample Output



184	4	12	18	22	36	47	55	75	77	79	80	84	84	84	104	116	127	1
769	756	350	601	522	334	84	301	80	340	575	22	638	223	1				

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Section 2 - Essay Writing

Section Summary

- No. of Questions: 1
- Duration: 15 min

Additional Instructions:

None

Q1. Essay Writing

Play is only fun when you win

Directions

write an essay for the given topic

Keywords



Answer Key & Solution

Section 1 - Coding Proficiency

Q1

Test Case

Input

```
10 1000
```

Output

```
153 370 371 407
```

Weightage - 20

Input

```
10 10000
```

Output

```
153 370 371 407 1634 8208 9474
```

Weightage - 25

Input

```
10 100000
```

Output

```
153 370 371 407 1634 8208 9474 54748 92727 93084
```

Weightage - 25

Input

```
10 1000000
```

Output

```
153 370 371 407 1634 8208 9474 54748 92727 93084
```

Weightage - 25

Input

```
500 700
```

Output

```
No elements
```

Weightage - 5

Sample Input

```
100 500
```

Sample Output

```
153 370 371 407
```

Sample Input

```
10 100
```

Sample Output

```
No elements
```

Solution



```

#include<stdio.h>
int main()
{
    int num,nod=0,power=1,flag =0 ,sum=0,digit,ctr,result,start,end;
    scanf("%d %d",&start,&end);
    for(num = start ; num < end ; num++,sum =0,nod=0)
    {
        //calculate nod
        power=1;
        while(num/power!=0)
        {
            nod++;
            power = power*10;
        }
        //reinitialize
        power=1;
        while(num/power!=0)
        {
            digit = (num/power)%10;
            for(ctr=1,result=1 ; ctr <= nod ; ctr++)
                result = result * digit;
            sum = sum + result;
            power = power * 10;
        }
        if(num == sum)
        {
            printf("%d ",num);
            flag = 1;
        }
    }
    if(flag == 0 )
        printf("No elements");

    return 0;
}

```

```

#include<stdio.h>
int main()
{
    int num,nod=0,power=1,flag =0 ,sum=0,digit,ctr,result,start,end;
    scanf("%d %d",&start,&end);
    for(num = start ; num < end ; num++,sum =0,nod=0)
    {
        //calculate nod
        power=1;
        while(num/power!=0)
        {
            nod++;
            power = power*10;
        }
        //reinitialize
        power=1;
        while(num/power!=0)
        {
            digit = (num/power)%10;
            for(ctr=1,result=1 ; ctr <= nod ; ctr++)
                result = result * digit;
            sum = sum + result;
            power = power * 10;
        }
        if(num == sum)
        {

```

```
        printf("%d ",num);
        flag = 1;
    }
}
if(flag == 0 )
    printf("No elements");

return 0;
}
```

Q2

Test Case

Input

Output

5
5 1 4 7 9

4 5 9

Weightage - 5

Input

Output

3
1 2 3

1 3

Weightage - 5

Input

Output

759
860 804 734 848 835 384 186 956 553 429 75 379 1

1 4 9 13 13 18 20 20 21 23 28 33 34 34 37 39 47

Weightage - 10

Input

Output

79
488 918 437 231 613 967 953 527 302 225 675 13 5

18 50 55 90 111 128 163 185 197 238 302 407 408

Weightage - 10

Input

Output

764
458 215 71 404 565 627 46 234 631 816 962 795 44

0 1 4 5 8 14 16 21 25 30 34 35 37 39 42 44 46 4

Weightage - 10

Input

Output

218
102 877 84 55 966 325 631 513 179 955 183 814 55

4 34 36 49 60 60 75 84 85 102 113 123 129 143 15



Weightage - 10

Input

```
913
514 108 551 149 74 56 36 942 416 221 176 679 714
```

Output

```
0 2 4 7 7 10 12 14 16 21 31 34 36 36 38 39 40 4
```

Weightage - 10

Input

```
176
523 378 265 820 929 466 24 931 353 1 610 711 955
```

Output

```
0 24 61 107 128 134 178 197 198 211 222 226 239
```

Weightage - 10

Input

```
946
615 713 611 57 832 952 734 64 584 277 566 989 52
```

Output

```
0 1 1 1 2 5 7 10 12 13 14 15 15 18 18 19 20 21
```

Weightage - 10

Input

```
387
801 759 329 385 315 113 987 84 80 722 530 582 41
```

Output

```
2 5 8 11 12 14 16 18 22 26 26 29 34 40 51 55 58
```

Weightage - 10

Input

```
980
328 249 354 686 938 656 116 765 181 802 783 942
```

Output

```
1 7 9 11 13 20 21 22 22 23 29 32 32 32 32 37 39
```

Weightage - 10

Sample Input

```
10
7 3 1 8 4 9 5 6 2 10
```

Sample Output

```
1 2 4 5 7
```

Sample Input

```
184
769 756 350 601 522 334 84 301 80 340 575 22 638
```

Sample Output

```
4 12 18 22 36 47 55 75 77 79 80 84 84 84 104 11
```

Solution

```
#include<stdio.h>
#include<limits.h>
```

```
#include<stdio.h>
#include<limits.h>
```



```
int main()

{
    int arr[1000],ctr,ctr1,min=INT_MAX,minPos=-1;
    int size,temp;
    scanf("%d",&size);
    for( ctr = 0 ; ctr< size; ctr++ )
        scanf("%d",&arr[ctr]);
    for(ctr =0 ; ctr< size -2 ; ctr+=2)
    {
        for( ctr1 = ctr ; ctr1 < size ; ctr1+=2 )
        {
            if( arr[ctr1] < min )
            {
                min = arr[ctr1];
                minPos = ctr1;
            }
        }
        temp = arr[ctr];
        arr[ctr] = arr[minPos];
        arr[minPos] = temp;
        min = INT_MAX;
        minPos = -1;
    }
    for( ctr = 0; ctr< size ; ctr+=2)
        printf("%d ", arr[ctr]);
}
```

```
int main()

{
    int arr[1000],ctr,ctr1,min=INT_MAX,minPos=-1;
    int size,temp;
    scanf("%d",&size);
    for( ctr = 0 ; ctr< size; ctr++ )
        scanf("%d",&arr[ctr]);
    for(ctr =0 ; ctr< size -2 ; ctr+=2)
    {
        for( ctr1 = ctr ; ctr1 < size ; ctr1+=2 )
        {
            if( arr[ctr1] < min )
            {
                min = arr[ctr1];
                minPos = ctr1;
            }
        }
        temp = arr[ctr];
        arr[ctr] = arr[minPos];
        arr[minPos] = temp;
        min = INT_MAX;
        minPos = -1;
    }
    for( ctr = 0; ctr< size ; ctr+=2)
        printf("%d ", arr[ctr]);
}
```

Section 2 - Essay Writing

Q1

Sample Essay

No Essay

Keywords

play, fun, win,

