RAJALAKSHMI ENGINEERING COLLEGE

An Autonomous Institution
Affiliated to Anna University, Chennai,
Rajalakshmi Nagar, Thandalam – 602 105

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

CS23331 – Design And Analysis of Algorithms
Laboratory Record Note Book

Name: VIJAYANAND.J……………………………………………………

Register No. : 231501181…………………………………………………………..

Year / Branch / Section: 2nd Year/AIML/C

Semester: III

Academic Year: 2024-2025

RAJALAKSHMI ENGINEERING COLLEGE
An Autonomous Institution
Affiliated to Anna University, Chennai,
Rajalakshmi Nagar, Thandalam – 602 105

BONAFIDE CERTIFICATE

Name: VIJAYANAND.J……………………………

Academic Year: 2024-2025…… Semester: III…… Branch: B.Tech- AIML….

Register No. 231501181

Certified that this is the bonafide record of work done by the above student in

the CS23331- Design And Analysis of Algorithms............................... Laboratory

during the academic year 2024- 2025

Signature of Faculty in-charge

Submitted for the Practical Examination held on   22/11/2024…………

Internal Examiner   **External Examiner**

<u>INDEX</u>

S.NO	Date	Name of the Experiment
1	10/8/24	Basic C Programming
2	1/9/24	Finding Time Complexity Of Algorithms
3	15/9/24	Divide and Conquer
4	24/9/24	Greedy Algorithms
5	5/11/24	Dynamic Programming
6	7/11/24	Competitive Programming

Started on	Saturday, 10 August 2024, 1:55 PM
State	Finished
Completed on	Saturday, 10 August 2024, 2:47 PM
Time taken	52 mins 23 secs
Marks	15.00/15.00
Grade	**100.00** out of 100.00

Given two numbers, write a C program to swap the given numbers.

For example:

Input	Result
10 20	20 10

Answer: (penalty regime: 0 %)

```
1   #include<stdio.h>
2   int main()
3 ▾ {
4       int a,b,c;
5       scanf("%d %d",&a,&b);
6       c=a;
7       a=b;
8       b=c;
9       printf("%d %d",a,b);
10  }
```

	Input	Expected	Got	
✔	10 20	20 10	20 10	✔

Passed all tests! ✔

Marks for this submission: 1.00/1.00.

2nd Year/AIML/C

Write a C program to find the eligibility of admission for a professional course based on the following criteria:

Marks in Maths >= 65

Marks in Physics >= 55

Marks in Chemistry >= 50

Or

Total in all three subjects >= 180

Sample Test Cases

Test Case 1

Input

 70 60 80

Output

The candidate is eligible

Test Case 2

Input

50 80   80

Output

The candidate is eligible

Test Case 3
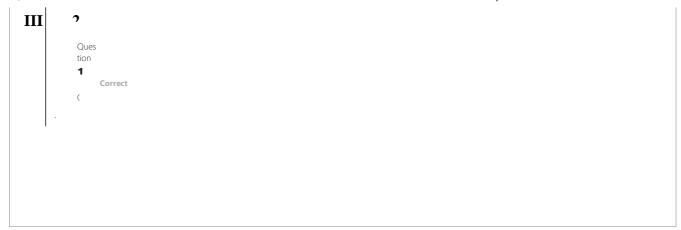
Input

50 60   40

Output

The candidate is not eligible

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  int main()
3  {
4      int a,b,c;
5      scanf("%d %d %d",&a,&b,&c);
6      if((a>=65 && b>=55 && c>=50) || (a+b+c)>=180)
```

III   **?**

Ques
tion
1
          Correct

(

.

{

Passed all tests! ✔

printf
Marks for this submission: 1.00/1.00.

```
}
else
{

```

Malini goes to BestSave hyper market to buy grocery items. BestSave hyper market provides 10% discount on the bill amount B when ever the bill amount B is more than Rs.2000.

The bill amount B is passed as the input to the program. The program must print the final amount A payable by Malini.

Input Format:

The first line denotes the value of B.

Output Format:

The first line contains the value of the final payable amount A.

Example Input/Output 1:

Input:

1900

Output:

1900

Example Input/Output 2:

Input:

3000

Output:

2700

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  int main()
3  {
4      int s;
5      scanf("%d",&s);
6      if(s<=2000)
7      {
8          printf("%d",s);
9      }
10     else
11     {
12         int d= s*0.1;
13         s=s-d;
14         printf("%d",s);
15     }
16 }
```

	Input	Expected	Got	
✔	1900	1900	1900	✔
✔	3000	2700	2700	✔

Passed all tests! ✔

printf
Marks for this submission: 1.00/1.00.

```
}
```

Baba is very kind to beggars and every day Baba donates half of the amount he has when ever a beggar requests him. The money M left in Baba's hand is passed as the input and the number of beggars B who received the alms are passed as the input. The program must print the money Baba had in the beginning of the day.

Input Format:

The first line denotes the value of M.
The second line denotes the value of B.

Output Format:

The first line denotes the value of money with Baba in the beginning of the day.

Example Input/Output:

Input:

100
2

Output:

400

Explanation:

Baba donated to two beggars. So when he encountered second beggar he had 100*2 = Rs.200 and when he encountered 1st he had 200*2 = Rs.400.

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a,b;
5      scanf("%d %d",&a,&b);
6      for(int i=0;i<b;i++)
7      {
8          a*=2;
9      }
10     printf("%d",a);
11 }
```

	Input	Expected	Got	
✔	100 2	400	400	✔

Passed all tests! ✔

Marks for this submission: 1.00/1.00.

	Input	Expected	Got	
✔	70 60 80	The candidate is eligible	The candidate is eligible	✔

✔   ✔

The CEO of company ABC Inc wanted to encourage the employees coming on time to the office. So he announced that for every consecutive day an employee comes on time in a week (starting from Monday to Saturday), he will be awarded Rs.200 more than the previous day as "Punctuality Incentive". The incentive I for the starting day (ie on Monday) is passed as the input to the program. The number of days N an employee came on time consecutively starting from Monday is also passed as the input. The program must calculate and print the "Punctuality Incentive" P of the employee.

Input Format:

The first line denotes the value of I.
The second line denotes the value of N.

Output Format:

The first line denotes the value of P.

Example Input/Output:

Input:

500
3

Output:

2100

Explanation:

On Monday the employee receives Rs.500, on Tuesday Rs.700, on Wednesday Rs.900

So total = Rs.2100

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a,b,sum=0;
5      scanf("%d %d",&a,&b);
6      for(int i=0;i<b;i++)
7      {
8          sum+=a;
9          a+=200;
10     }
11     printf("%d",sum);
12 }
```

	Input	Expected	Got	
✔	500 3	2100	2100	✔
✔	100 3	900	900	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Two numbers M and N are passed as the input. A number X is also passed as the input. The program must print the numbers divisible by X from N to M (inclusive of M and N).

Input Format:

The first line denotes the value of M
The second line denotes the value of N
The third line denotes the value of X

Output Format:

Numbers divisible by X from N to M, with each number separated by a space.

Boundary Conditions:

$1 <= M <= 9999999$
$M < N <= 9999999$
$1 <= X <= 9999$

Example Input/Output 1:

Input:
2
40
7

Output:
35 28 21 14 7

Example Input/Output 2:

Input:
66
121
11

Output:
121 110 99 88 77 66

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d %d %d",&a,&b,&c);
    for(int i=b;i>=a;i--)
    {
        if(i%c==0)
        {
            printf("%d ",i);
        }
    }
}
```

	Input	Expected	Got	
✔	2 40 7	35 28 21 14 7	35 28 21 14 7	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

Write a C program to find the quotient and reminder of given integers.

For example:

Input	Result
12 3	4 0

Answer: (penalty regime: 0 %)

```
1   #include<stdio.h>
2   int main()
3 ▾ {
4       int a,b;
5       scanf("%d %d",&a,&b);
6       int c= a/b;
7       int d=a%b;
8       printf("%d\n%d",c,d);
9   }
```

	Input	Expected	Got	
✔	12 3	4 0	4 0	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

Question **12**

Correct

Mark 1.00 out of 1.00

Write a C program to find the biggest among the given 3 integers?

For example:

Input	Result
10 20 30	30

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a,b,c;
5      scanf("%d %d %d",&a,&b,&c);
6      if(a>b && a>c)
7      {
8          printf("%d",a);
9      }
10     else if(b>a && b>c)
11     {
12         printf("%d",b);
13     }
14     else
15     {
16         printf("%d",c);
17     }
18 }
```

	Input	Expected	Got	
✔	10 20 30	30	30	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

Question **13**

Correct

Mark 1.00 out of 1.00

Write a C program to find whether the given integer is odd or even?

For example:

Input	Result
12	Even
11	Odd

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a;
5      scanf("%d",&a);
6      if(a%2==0)
7      {
8          printf("Even");
9      }
10     else
11     {
12         printf("Odd");
13     }
14 }
```

	Input	Expected	Got	
✔	12	Even	Even	✔
✔	11	Odd	Odd	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **14**

Correct

Mark 1.00 out of 1.00

Write a C program to find the factorial of given n.

For example:

Input	Result
5	120

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a;
5      scanf("%d",&a);
6      int sum=1;
7      for(int i=1;i<=a;i++)
8      {
9          sum*=i;
10     }
11     printf("%d",sum);
12  }
```

	Input	Expected	Got	
✔	5	120	120	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

Question **15**

Correct

Mark 1.00 out of 1.00

Write a C program to find the sum first N natural numbers.

For example:

Input	Result
3	6

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a,sum=0;
5      scanf("%d",&a);
6      for(int i=1;i<=a;i++)
7      {
8          sum+=i;
9      }
10     printf("%d",sum);
11 }
```

	Input	Expected	Got	
✔	3	6	6	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **16**

Correct

Mark 1.00 out of 1.00

Write a C program to find the Nth term in the fibonacci series.

For example:

Input	Result
0	0
1	1
4	3

Answer: (penalty regime: 0 %)

```c
1   #include<stdio.h>
2   int main()
3   {
4       int n,a=0,b=1,c,i;
5       scanf("%d",&n);
6       if(n==0)
7       {
8           printf("0");
9       }
10      else
11      {
12          for(i=2;i<=n;i++)
13          {
14              c=a+b;
15              a=b;
16              b=c;
17          }
18          printf("%d",b);
19      }
20
21  }
```

	Input	Expected	Got	
✔	0	0	0	✔
✔	1	1	1	✔
✔	4	3	3	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

Question **17**

Correct

Mark 1.00 out of 1.00

Write a C program to find the power of integers.

input:

a b

output:

a^b value

For example:

Input	Result
2 5	32

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  #include<math.h>
3  int main()
4  {
5      int a,b;
6      scanf("%d %d",&a,&b);
7      int c=pow(a,b);
8      printf("%d",c);
9  }
```

	Input	Expected	Got	
✔	2 5	32	32	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **18**

Correct

Mark 1.00 out of 1.00

Write a C program to find Whether the given integer is prime or not.

For example:

Input	Result
7	Prime
9	No Prime

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,flag=1;
    scanf("%d",&a);
    for(int i=2;i<a/2;i++)
    {
        if(a%i==0)
        {
            printf("No Prime");
            flag=0;
            break;
        }
    }
    if(flag==1)
    {
        printf("Prime");
    }
}
```

	Input	Expected	Got	
✔	7	Prime	Prime	✔
✔	9	No Prime	No Prime	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Question **19**

Correct

Mark 1.00 out of 1.00

Write a C program to find the reverse of the given integer?

Answer: (penalty regime: 0 %)

```
 1  #include<stdio.h>
 2  int main()
 3  {
 4      int a,reverse=0,rem;
 5      scanf("%d",&a);
 6      while(a!=0)
 7      {
 8          rem=a%10;
 9          reverse=reverse*10+rem;
10          a/=10;
11      }
12      printf("%d",reverse);
13
14  }
```

	Input	Expected	Got	
✔	123	321	321	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ DAA Model Exam 2024

Correct

Problem 1: Finding Complexity using Counter Method ►

Started on	Saturday, 14 September 2024, 1:37 PM
State	Finished
Completed on	Saturday, 14 September 2024, 1:49 PM
Time taken	11 mins 25 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Correct

Correct

Correct

Correct

For example:

Input	Result
9	12

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
void function (int n)
{
    int count=0;
    int i= 1;
    count++;
    int s =1;
    count++;
    while(s <= n)
    {
        count++;
        i++;
        count++;
        s += i;
        count++;
    }
     count++;
     printf("%d",count);
}
int main()
{
    int b;
    scanf("%d",&b);
    function(b);
}
```

	Input	Expected	Got	
✔	9	12	12	✔
✔	4	9	9	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ BASIC C PROGRAMMING-PRACTICE

Jump to...

Problem 2: Finding Complexity using Counter method ►

Marks for this submission: 1.00/1.00.

◄ BASIC C PROGRAMMING-PRACTICE

Jump to...

Problem 2: Finding Complexity using Counter method ►

Started on	Saturday, 9 November 2024, 2:19 PM
State	Finished
Completed on	Saturday, 9 November 2024, 2:24 PM
Time taken	4 mins 56 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Convert the following algorithm into a program and find its time complexity using the counter method.
```
void function (int n)
{
    int i= 1;
```

Answer: (penalty regime: 0 %)

```
1   #include<stdio.h>
2   #include<stdlib.h>
3   int main()
4   {
5       int n;
6       scanf("%d",&n);
7       int c=0;
8       if(n==1)
9       {
10          c++;
11      }
12      else
13      {
14
15          for(int i=1; i<=n; i++)
16          {
17              c++;
18              c++;
19              for(int j=1; j<=n; j++)
20              {
21                  c++;
22                  c++;
23                  break;
24              }
25              c++;
26          }
27          c++;
28      }
29      c++;
30      printf("%d",c);
31      return 0;
32  }
33
```

	Input	Expected	Got	
✔	2	12	12	✔
✔	1000	5002	5002	✔
✔	143	717	717	✔

Passed all tests! ✔

in

Marks for this submission: 1.00/1.00.

◄ Problem 1: Finding Complexity using Counter Method

```
while(s <= n)
{
```

Problem 3: Finding Complexity using Counter Method ►

Started on	Saturday, 9 November 2024, 1:49 PM
State	Finished
Completed on	Saturday, 9 November 2024, 1:57 PM
Time taken	7 mins 58 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Correct

Jump to...

Answer:

```c
1  #include<stdio.h>
2  #include<stdlib.h>
3  int main(){
4      int n;
5      scanf("%d",&n);
6      int counter=0;
7      for(int i=1;i<=n;++i)
8      {
9          counter++;
10         counter++;
11         if(n%i==0)
12         {
13             counter++;
14         }
15     }
16     counter++;
17     printf("%d",counter);
18     return 0;
19 }
```

	Input	Expected	Got	
✔	12	31	31	✔
✔	25	54	54	✔
✔	4	12	12	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

◄ Problem 2: Finding Complexity using Counter method

Convert the following algorithm into a program and find its time complexity using the counter method.
void func(int n)

Started on	Saturday, 9 November 2024, 1:58 PM
State	Finished
Completed on	Saturday, 9 November 2024, 2:12 PM
Time taken	13 mins 37 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Correct

Jump to

Question **1**

Correct

Mark 1.00 out of 1.00

Answer:

```c
1  #include<stdio.h>
2  #include<stdlib.h>
3  int main()
4  {
5      int n;
6      scanf("%d",&n);
7      int count=0;
8      int c=0;
9      for(int i=n/2;i<n;i++)
10     {
11         count++;
12         for(int j=1;j<n;j=2*j)
13         {
14             count++;
15             for(int k=1;k<n;k=k*2)
16             {
17                 count++;
18                 count++;
19                 c++;
20             }
21             count++;
22         }
23         count++;
24     }
25     count++;
26     count++;
27     printf("%d",count);
28     return 0;
29 }
```

	Input	Expected	Got	
✔	4	30	30	✔
✔	10	212	212	✔

Passed all tests! ✔

c

Marks for this submission: 1.00/1.00.

Correct

Correct

Started on	Saturday, 9 November 2024, 2:24 PM
State	Finished
Completed on	Saturday, 9 November 2024, 2:27 PM
Time taken	3 mins 27 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Jump to...

Question **1**

Correct

Mark 1.00 out of 1.00

Answer:

```c
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int n;
    scanf("%d",&n);
    int c=0;
    int rev = 0, remainder;
    while (n != 0)
    {
        c++;
        remainder = n % 10;
        rev = rev * 10 + remainder;
        n/= 10;
        c++;
        c++;
        c++;

    }
    c++;
    c++;
    c++;
    printf("%d",c);
    return 0;
}
```

	Input	Expected	Got	
✔	12	11	11	✔
✔	1234	19	19	✔

Passed all tests! ✔

Conver

Marks for this submission: 1.00/1.00.

complexity using counter method.

complexity using counter method.

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 1-Number of Zeros in a Given Array

Started on	Sunday, 10 November 2024, 12:02 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:08 PM
Time taken	6 mins 21 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Correct

Problem Statement

Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program using Divide and Conquer to Count the number of zeroes in the given array.

Input Format

   First Line Contains Integer m – Size of array

   Next m lines Contains m numbers – Elements of an array

Output Format

   First Line Contains Integer – Number of zeroes present in the given array.

Answer: (penalty regime: 0 %)

```c
1   #include<stdio.h>
2   int main()
3   {
4       int n;
5       scanf("%d",&n);
6       int arr[n];
7       int c=0;
8       for(int i=0;i<n;i++)
9       {
10          scanf("%d",&arr[i]);
11      }
12      for(int i=0;i<n;i++)
13      {
14          if(arr[i]==0)
15          {
16              c++;
17          }
18      }
19      printf("%d",c);
20      return 0;
21  }
```

	Input	Expected	Got	
✔	5 1 1 1 0 0	2	2	✔
✔	10 1 1 1 1 1 1 1 1 1 1	0	0	✔

	Input	Expected	Got	
✔	8 0 0 0 0 0 0 0 0	8	8	✔
✔	17 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0	2	2	✔

Passed all tests! ✔

Jump

Marks for this submission: 1.00/1.00.

◄ Problem 5: Finding Complexity using counter method

Question **1**

2-Majority Element ►

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 2-Majority Element

Started on	Sunday, 10 November 2024, 12:04 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:11 PM
Time taken	6 mins 9 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 2-Majority Element

```
Convert the following algorithm into a program and find its time complexity using counter method.
void reverse(int n)
{
   int rev = 0. remainder:
```

Given an array `nums` of size `n`, return *the majority element*.

The majority element is the element that appears more than $\lfloor n / 2 \rfloor$ times. You may assume that the majority element always exists in the array.

Example 1:

> Correct

Example 2:

> Jump to…

Constraints:

- `n == nums.length`
- `1 <= n <= 5 * 10⁴`
- `-2³¹ <= nums[i] <= 2³¹ - 1`

For example:

Input	Result
3 3 2 3	3
7 2 2 1 1 1 2 2	2

Answer: (penalty regime: 0 %)

```c
1   #include<stdio.h>
2   int main()
3   {
4       int n;
5       scanf("%d",&n);
6       int arr[n];
7       for(int i=0;i<n;i++)
8       {
9           scanf("%d",&arr[i]);
10      }
11      int maxElement=100;
12      int frequency[maxElement+1];
13      for(int i=0;i<=maxElement;i++)
14      {
15          frequency[i]=0;
16      }
17      for(int i=0;i<n;i++)
18      {
19          frequency[arr[i]]++;
20      }
21      for(int i=0;i<=maxElement;i++)
22      {
23          if(frequency[i]>n/2)
24          {
25              printf("%d",i);
26          }
27      }
28  }
```

	Input	Expected	Got	
✔	3 3 2 3	3	3	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

◄ 1-Number of Zeros in a Given Array

Correct

3-Finding Floor Value ►

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 3-Finding Floor Value

Started on	Sunday, 10 November 2024, 12:09 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:14 PM
Time taken	5 mins 26 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 3-Finding Floor Value

Jump to...

Problem Statement:

Given a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

Input Format

   First Line Contains Integer n – Size of array

   Next n lines Contains n numbers – Elements of an array

   Last Line Contains Integer x – Value for x

Output Format

   First Line Contains Integer – Floor value for x

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int n,x;
5      scanf("%d",&n);
6      int arr[n];
7      for(int i=0;i<n;i++)
8      {
9          scanf("%d",&arr[i]);
10     }
11     scanf("%d",&x);
12     int minx=x;
13     for(int i=0;i<n;i++)
14     {
15         if(arr[i]<=x)
16         {
17             minx=arr[i];
18         }
19     }
20     printf("%d",minx);
21     return 0;
22 }
```

	Input	Expected	Got	
✔	6 1 2 8 10 12 19 5	2	2	✔
✔	5 10 22 85 108 129 100	85	85	✔

	Input	Expected	Got	
✔	7 3 5 7 9 11 13 15 10	9	9	✔

Passed all tests! ✔

Questio

Marks for this submission: 1.00/1.00.

◄ 2-Majority Element

```
Input: nums = [3,2,3]
Output: 3
```

4-Two Elements sum to x ►

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 4-Two Elements sum to x

Started on	Sunday, 10 November 2024, 12:11 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:21 PM
Time taken	10 mins 5 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 4-Two Elements sum to x

```
Input: nums = [2,2,1,1,1,2,2]
Output: 2
```

Problem Statement:

Given a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return the numbers, otherwise print as "No".

Note: Write a Divide and Conquer Solution

Input Format

   First Line Contains Integer n – Size of array

   Next n lines Contains n numbers – Elements of an array

   Last Line Contains Integer x – Sum Value

Output Format

   First Line Contains Integer – Element1

   Second Line Contains Integer – Element2 (Element 1 and Elements 2 together sums to value "x")

Answer: (penalty regime: 0 %)

```c
 1  #include<stdio.h>
 2  int main()
 3  {
 4      int n;
 5      scanf("%d",&n);
 6      int arr[n];
 7      int found=0;
 8      for(int i=0;i<n;i++)
 9      {
10          scanf("%d",&arr[i]);
11      }
12      int sum;
13      scanf("%d",&sum);
14      for(int i=0;i<n;i++)
15      {
16          for(int j=i+1;j<n;j++)
17          {
18              if(arr[i]+arr[j]==sum)
19              {
20                  found=1;
21                  printf("%d\n",arr[i]);
22                  printf("%d",arr[j]);
23                  break;
24              }
25          }
26      }
27      if(found==0)
28      {
29          printf("No");
30      }
31      return 0;
32  }
```

	Input	Expected	Got	
✔	4 2 4 8 10 14	4 10	4 10	✔
✔	5 2 4 6 8 10 100	No	No	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

◄ 3-Finding Floor Value

Jump to...

Dashboard / My courses / CS23331-DAA-2023-AIML / Divide and Conquer / 6-Implementation of Quick Sort

Started on	Sunday, 10 November 2024, 12:14 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:21 PM
Time taken	6 mins 52 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Write a Program to Implement the Quick Sort Algorithm

Input Format:

The first line contains the no of elements in the list-n

The next n lines contain the elements.

Output:

Sorted list of elements

For example:

Input	Result
5 67 34 12 98 78	12 34 67 78 98

Answer:

```c
#include<stdio.h>
void swap(int arr[],int i,int j)
{
    int temp=arr[i];
    arr[i]=arr[j];
    arr[j]=temp;
}
int partition(int arr[],int low,int high)
{
    int pivot=arr[high];
    int i=low-1;
    for(int j=low;j<high;j++)
    {
        if(arr[j]<pivot)
        {
            i++;
            swap(arr,i,j);
        }
    }
    swap(arr,i+1,high);
    return(i+1);
}
void quickSort(int arr[],int low,int high)
{
    if(low<high)
    {
        int pi=partition(arr,low,high);
        quickSort(arr,low,pi-1);
        quickSort(arr,pi+1,high);
    }
}
int main()
{
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    quickSort(arr,0,n-1);
    for(int i=0;i<n;i++)
    {
        printf("%d ",arr[i]);
    }
    return 0;
}
```

	Input	Expected	Got	
✔	5 67 34 12 98 78	12 34 67 78 98	12 34 67 78 98	✔
✔	10 1 56 78 90 32 56 11 10 90 114	1 10 11 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	✔
✔	12 9 8 7 6 5 4 3 2 1 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

◄ 4-Two Elements sum to x

Jump to...

1-G-Coin Problem ►

Started on	Monday, 30 September 2024, 2:07 PM
State	Finished
Completed on	Monday, 30 September 2024, 2:09 PM
Time taken	2 mins 19 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Write a program to take value V and we want to make change for V Rs, and we have infinite supply of each of the denominations in Indian currency, i.e., we have infinite supply of { 1, 2, 5, 10, 20, 50, 100, 500, 1000} valued coins/notes, what is the minimum number of coins and/or notes needed to make the change.

Input Format:

Take an integer from stdin.

Output Format:

print the integer which is change of the number.

Example Input :

64

Output:

4

Explanaton:

We need a 50 Rs note and a 10 Rs note and two 2 rupee coins.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
void mincoincount(int x)
{
    int coins[] = {1000, 500, 100, 50, 20, 10, 5, 2, 1};
    int i = 0, count = 0;
    while (x > 0)
    {
        if (x >= coins[i])
        {
            x -= coins[i];
            count++;
        }
        else
        {i++;
        }
    }
    printf("%d\n", count);
}
int main() {
    int x;
    scanf("%d", &x);
    mincoincount(x);
    return 0;
}
```

	Input	Expected	Got	
✔	49	5	5	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 6-Implementation of Quick Sort

Jump to...

Dashboard / My courses / CS23331-DAA-2023-AIML / Greedy Algorithms / 2-G-Cookies Problem

Started on	Monday, 30 September 2024, 2:09 PM
State	Finished
Completed on	Monday, 30 September 2024, 2:10 PM
Time taken	55 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Greedy Algorithms / 2-G-Cookies Problem

Question **1**

Correct

Mark 1.00 out of 1.00

Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.

Each child i has a greed factor $g[i]$, which is the minimum size of a cookie that the child will be content with; and each cookie j has a size $s[j]$. If $s[j] >= g[i]$, we can assign the cookie j to the child i, and the child i will be content. Your goal is to maximize the number of your content children and output the maximum number.

Example 1:

Input:

3

1 2 3

2

1 1

Output:

1

Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1, 2, 3.

And even though you have 2 cookies, since their size is both 1, you could only make the child whose greed factor is 1 content.

You need to output 1.

Constraints:

$1 <= g.length <= 3 * 10^4$

$0 <= s.length <= 3 * 10^4$

$1 <= g[i], s[j] <= 2^{31} - 1$

Answer: (penalty regime: 0 %)

```c
1  #include <stdio.h>
2  int main() {
3      int a, b, i, j, e = 0;
4      scanf("%d", &a);
5      int x[a];
6      for(i=0; i< a; i++)
7      {
8          scanf("%d", &x[i]);
9      }
10     scanf("%d", &b);
11     int y[b];
12     for (i = 0; i < b; i++)
13     {
14         scanf("%d", &y[i]);
15     }
16     for (i = 1; i < a; i++)
17     {
18         for (j = i; j > 0 && x[j] < x[j - 1]; j--) {
19             int temp = x[j];
20             x[j] = x[j - 1];
21             x[j - 1] = temp;
22         }
23     }
24     for (i = 1; i < b; i++)
25     {
26         for (j = i; j > 0 && y[j] < y[j - 1]; j--) {
27             int temp = y[j];
28             y[j] = y[j - 1];
29             y[j - 1] = temp;
30         }
31     }
32     for (j = 0, i = 0; j < b; j++)
33     {
34         if (i < a && y[j] >= x[i])
35         {
36             e++;
37             i++;
38         }
```

```
39        }
40        printf("%d", e);
41
42        return 0;
43   }
```

	Input	Expected	Got	
✔	2 1 2 3 1 2 3	2	2	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 1-G-Coin Problem

Jump to...

3-G-Burger Problem ►

Started on	Sunday, 10 November 2024, 12:23 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:32 PM
Time taken	9 mins 7 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Correct

For example:

Test	Input	Result
Test Case 1	3 1 3 2	18

Answer: (penalty regime: 0 %)

```c
 1  #include<stdio.h>
 2  #include<math.h>
 3  int main()
 4  {
 5      int n,s=0;
 6      scanf("%d",&n);
 7      int cal[n];
 8      for(int i=0;i<n;i++)
 9      {
10          scanf("%d",&cal[i]);
11      }
12      for(int i=0;i<n-i-1;i++)
13      {
14          for(int j=0;j<n-i-1;j++)
15          {
16              if(cal[j]<cal[j+1])
17              {
18                  int temp=cal[j];
19                  cal[j]=cal[j+1];
20                  cal[j+1]=temp;
21              }
22          }
23      }
24      for(int i=0;i<n;i++)
25      {
26          s+=pow(n,i)*cal[i];
27      }
28      printf("%d",s);
29      return 0;
30  }
```

	Test	Input	Expected	Got	
✔	Test Case 1	3 1 3 2	18	18	✔
✔	Test Case 2	4 7 4 9 6	389	389	✔
✔	Test Case 3	3 5 10 7	76	76	✔

Passed all tests! ✔

Jump

Marks for this submission: 1.00/1.00.

◄ 2-G-Cookies Problem

Question **1**

Dashboard / My courses / CS23331-DAA-2023-AIML / Greedy Algorithms / 4-G-Array Sum max problem

Started on	Monday, 30 September 2024, 2:10 PM
State	Finished
Completed on	Monday, 30 September 2024, 2:11 PM
Time taken	47 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Greedy Algorithms / 4-G-Array Sum max problem

Correct

Given an array of N integer, we have to maximize the sum of arr[i] * i, where i is the index of the element (i = 0, 1, 2, ..., N).Write an algorithm based on Greedy technique with a Complexity O(nlogn).

 Input Format:

First line specifies the number of elements-n

The next n lines contain the array elements.

Output Format:

Maximum Array Sum to be printed.

Sample Input:

5

2 5 3 4 0

Sample output:

40

Answer: (penalty regime: 0 %)

```c
1  #include <stdio.h>
2  int main()
3  {
4      int n;
5      scanf("%d",&n);
6      int arr[n];
7      for (int i=0;i<n;i++)
8      {
9          scanf("%d",&arr[i]);
10     }
11     for (int i=0;i<n-1;i++)
12     {
13         for (int j=0;j<n-i-1;j++)
14         {
15             if(arr[j]>arr[j + 1])
16             {
17                 int temp = arr[j];
18                 arr[j] = arr[j + 1];
19                 arr[j + 1] = temp;
20             }
21         }
22     }
23     int max_sum = 0;
24     for (int i=0;i<n;i++)
25     {
26         max_sum += arr[i] * i;
27     }
28     printf("%d\n",max_sum);
29     return 0;
30 }
```

	Input	Expected	Got	
✔	5 2 5 3 4 0	40	40	✔

	Input	Expected	Got	
✔	10 2 2 2 4 4 3 3 5 5 5	191	191	✔
✔	2 45 3	45	45	✔

Passed all tests! ✔

Jump

Marks for this submission: 1.00/1.00.

◄ 3-G-Burger Problem

Question **1**

5-G-Product of Array elements-Minimum ►

Started on	Monday, 30 September 2024, 2:11 PM
State	Finished
Completed on	Monday, 30 September 2024, 2:12 PM
Time taken	41 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

A person needs to eat burgers. Each burger contains a count of calorie. After eating the burger, the person needs to run a distance to burn out his calories.

If he has eaten i burgers with c calories each, then he has to run at least $3^i * c$ kilometers to burn out the calories. For example, if he ate 3

Given two arrays array_One[] and array_Two[] of same size N. We need to first rearrange the arrays such that the sum of the product of pairs(1 element from each) is minimum. That is SUM (A[i] * B[i]) for all i is minimum.

For example:

Input	Result
3 1 2 3 4 5 6	28

Answer: (penalty regime: 0 %)

```c
1   #include <stdio.h>
2   #include <stdlib.h>
3   int main()
4   {
5       int N;
6       scanf("%d",&N);
7   
8       int array_One[N], array_Two[N];
9   
10      for (int i=0;i<N;i++)
11      {
12          scanf("%d",&array_One[i]);
13      }
14   
15      for (int i=0;i<N;i++)
16      {
17          scanf("%d",&array_Two[i]);
18      }
19   
20      for (int i=0;i<N-1;i++)
21      {
22          for (int j=0;j<N-i-1;j++)
23          {
24              if (array_One[j] > array_One[j + 1])
25              {
26                  int temp = array_One[j];
27                  array_One[j] = array_One[j + 1];
28                  array_One[j + 1] = temp;
29              }
30          }
31      }
32   
33      for (int i=0;i<N-1;i++)
34      {
35          for (int j=0;j<N-i-1;j++)
36          {
37              if (array_Two[j] > array_Two[j + 1])
38              {
39                  int temp = array_Two[j];
40                  array_Two[j] = array_Two[j + 1];
41                  array_Two[j + 1] = temp;
42              }
43          }
44      }
45   
46      int min_sum = 0;
47      for (int i=0;i<N;i++)
48      {
49          min_sum += array_One[i]*array_Two[N-i-1];
50      }
51      printf("%d\n",min_sum);
52      return 0;
```

	Input	Expected	Got	
✔	3 1 2 3 4 5 6	28	28	✔
✔	4 7 5 1 2 1 3 4 1	22	22	✔
✔	5 20 10 30 10 40 8 9 4 3 10	590	590	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 4-G-Array Sum max problem

Jump to...

1-DP-Playing with Numbers ►

Dashboard / My courses / CS23331-DAA-2023-AIML / Dynamic Programming / 1-DP-Playing with Numbers

Started on	Sunday, 10 November 2024, 12:46 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:56 PM
Time taken	9 mins 55 secs
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Dynamic Programming / 1-DP-Playing with Numbers

Question **1**

Correct

Mark 1.00 out of 1.00

Playing with Numbers:

Ram and Sita are playing with numbers by giving puzzles to each other. Now it was Ram term, so he gave Sita a positive integer 'n' and two numbers 1 and 3. He asked her to find the possible ways by which the number n can be represented using 1 and 3.Write any efficient algorithm to find the possible ways.

Example 1:

Input: 6
*Output:*6
Explanation: There are 6 ways to 6 represent number with 1 and 3

　　 1+1+1+1+1+1
　　 3+3
　　 1+1+1+3
　　 1+1+3+1
　　 1+3+1+1
　　 3+1+1+1

Input Format

First Line contains the number n

Output Format

Print: The number of possible ways 'n' can be represented using 1 and 3

Sample Input

6

Sample Output

6

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  #define MAX_N 100000
3  long long count(int n)
4  {
5      long long dp[n+1];
6      for(int i=0;i<=n;i++)
7      {
8          dp[i]=0;
9      }
10     dp[0]=1;
11     for(int i=1;i<=n;i++)
12     {
13         if(i>=1)
14         {
15             dp[i]+=dp[i-1];
16         }
17         if(i>=3)
18         {
19             dp[i]+=dp[i-3];
20         }
21     }
22     return dp[n];
23 }
24 int main()
25 {
26     int n;
27     scanf("%d",&n);
28     if(n<0)
29     {
30         return 1;
31     }
32     if(n>MAX_N)
33     {
```

```
34            return 1;
35        }
36    printf("%lld\n",count(n));
37    return 0;
38 }
```

	Input	Expected	Got	
✔	6	6	6	✔
✔	25	8641	8641	✔
✔	100	24382819596721629	24382819596721629	✔

Passed all tests! ✔

Correct
Marks for this submission: 10.00/10.00.

◄ 5-G-Product of Array elements-Minimum

Jump to...

2-DP-Playing with chessboard ►

Dashboard / My courses / CS23331-DAA-2023-AIML / Dynamic Programming / 2-DP-Playing with chessboard

Started on	Sunday, 10 November 2024, 12:50 PM
State	Finished
Completed on	Sunday, 10 November 2024, 12:59 PM
Time taken	9 mins 9 secs
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Playing with Chessboard:

Ram is given with an n*n chessboard with each cell with a monetary value. Ram stands at the (0,0), that the position of the top left white rook. He is been given a task to reach the bottom right black rook position (n-1, n-1) constrained that he needs to reach the position by traveling the maximum monetary path under the condition that he can only travel one step right or one step down the board. Help ram to achieve it by providing an efficient DP algorithm.

Example:

Input

3

1 2 4

2 3 4

8 7 1

Output:

19

Explanation:

Totally there will be 6 paths among that the optimal is
 Optimal path value:1+2+8+7+1=19

Input Format

First Line contains the integer n

The next n lines contain the n*n chessboard values

Output Format

Print Maximum monetary value of the path

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  #include<stdlib.h>
3  int max(int n,int chess[n][n])
4  {
5      int dp[n][n];
6      dp[0][0]=chess[0][0];
7      for(int j=1;j<n;j++)
8      {
9          dp[0][j]=dp[0][j-1]+chess[0][j];
10     }
11     for(int i=1;i<n;i++)
12     {
13         dp[i][0]=dp[i-1][0]+chess[i][0];
14     }
15     for(int i=1;i<n;i++)
16     {
17         for(int j=1;j<n;j++)
18         {
19             if(dp[i-1][j]>dp[i][j-1])
20             {
21                 dp[i][j]=dp[i-1][j]+chess[i][j];
22             }
23             else
24             {
25                 dp[i][j]=dp[i][j-1]+chess[i][j];
26             }
27         }
28     }
29     return dp[n-1][n-1];
30 }
31 int main()
32 {
33     int n;
34     scanf("%d",&n);
35     int chess[n][n];
36     for( int i=0;i<n;i++)
```

```
37 ▾    {
38          for(int j=0;j<n;j++)
39 ▾        {
40              scanf("%d",&chess[i][j]);
41          }
42      }
43      int result=max(n,chess);
44      printf("%d\n",result);
45      return 0;
46 }
```

	Input	Expected	Got	
✔	3 1 2 4 2 3 4 8 7 1	19	19	✔
✔	3 1 3 1 1 5 1 4 2 1	12	12	✔
✔	4 1 1 3 4 1 5 7 8 2 3 4 6 1 6 9 0	28	28	✔

Passed all tests! ✔

Correct
Marks for this submission: 10.00/10.00.

◄ 1-DP-Playing with Numbers

Jump to...

3-DP-Longest Common Subsequence ►

Started on	Sunday, 10 November 2024, 12:56 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:04 PM
Time taken	7 mins 47 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 10.00 out of 10.00

Given two strings find the length of the common longest subsequence(need not be contiguous) between the two.

Example:

s1: ggtabe

s2: tgatasb

s1		a	g	g	t	a	b	
s2	g	x		t	x	a	y	b

The length is 4

Solveing it using Dynamic Programming

For example:

Input	Result
aab azb	2

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
#include<stdlib.h>
int main()
{
    char a[11],b[11];
    scanf("%10s",a);
    scanf("%10s",b);
    int n=0,count=0;
    for(int i=0;i<sizeof(a);i++)
    {
        if(a[i]=='\0')
        {
            break;
        }
        else
        {
            n++;
        }
    }
    for(int i=0;i<n;i++)
    {
        if(a[i]==b[i])
        {
            count++;
        }
    }
    printf("%d",count);
}
```

	Input	Expected	Got	
✔	aab azb	2	2	✔
✔	ABCD ABCD	4	4	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Jump to...

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Dashboard / My courses / CS23331-DAA-2023-AIML / Dynamic Programming / 4-DP-Longest non-decreasing Subsequence

Started on	Sunday, 10 November 2024, 1:00 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:05 PM
Time taken	5 mins 20 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Dashboard / My courses / CS23331-DAA-2023-AIML / Dynamic Programming / 4-DP-Longest non-decreasing Subsequence

Question **1**

Correct

Mark 10.00 out of 10.00

Problem statement:

Find the length of the Longest Non-decreasing Subsequence in a given Sequence.

Eg:

Input:9

Sequence:[-1,3,4,5,2,2,2,2,3]

the subsequence is [-1,2,2,2,2,3]

Output:6

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int longest(int arr[],int n)
{
    int l[n];
    for(int i=0;i<n;i++)
    {
        l[i]=1;
    }
    for(int i=1;i<n;i++)
    {
        for(int j=0;j<i;j++)
        {
            if(arr[i]>=arr[j]&&l[i]<l[j]+1)
            {
                l[i]=l[j]+1;
            }
        }
    }
    int max_len=0;
    for(int i=0;i<n;i++)
    {
        if(l[i]>max_len)
        {
            max_len=l[i];
        }
    }
    return max_len;
}
int main()
{
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    int length=longest(arr,n);
    printf("%d\n",length);
    return 0;
}
```

	Input	Expected	Got	
✔	9 -1 3 4 5 2 2 2 2 3	6	6	✔
✔	7 1 2 2 4 5 7 6	6	6	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

◄ 3-DP-Longest Common Subsequence

Jump to...

Correct

Marks for this submission: 1.00/1.00.

◄ 3-DP-Longest Common Subsequence

Jump to...

Started on	Sunday, 10 November 2024, 1:05 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:16 PM
Time taken	10 mins 31 secs
Marks	1.00/1.00
Grade	**4.00** out of 4.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5 1 1 2 3 4	1

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    int  check[50];
    for(int i=0;i<50;i++)
    {
        check[i]=0;
    }
    for(int i=0;i<n;i++)
    {
        int a=arr[i];
        check[a]++;
    }
    for(int i=0;i<50;i++)
    {
        if(check[i]>1)
        {
            printf("%d",i);
        }
    }
}
```

	Input	Expected	Got	
✔	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✔
✔	5 1 2 3 4 4	4	4	✔
✔	5 1 1 2 3 4	1	1	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Jump to...

Started on	Sunday, 10 November 2024, 1:13 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:23 PM
Time taken	9 mins 17 secs
Marks	1.00/1.00
Grade	**4.00** out of 4.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5 1 1 2 3 4	1

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int n;
5      scanf("%d",&n);
6      int arr[n];
7      for(int i=0;i<n;i++)
8      {
9          scanf("%d",&arr[i]);
10     }
11     int  check[50];
12     for(int i=0;i<50;i++)
13     {
14         check[i]=0;
15     }
16     for(int i=0;i<n;i++)
17     {
18         int a=arr[i];
19         check[a]++;
20     }
21     for(int i=0;i<50;i++)
22     {
23         if(check[i]>1)
24         {
25             printf("%d",i);
26         }
27     }
28 }
```

	Input	Expected	Got	
✔	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✔
✔	5 1 2 3 4 4	4	4	✔
✔	5 1 1 2 3 4	1	1	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Jump to...

Started on	Sunday, 10 November 2024, 1:16 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:26 PM
Time taken	9 mins 53 secs
Marks	1.00/1.00
Grade	**30.00** out of 30.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

Input Format

· The first line contains T, the number of test cases. Following T lines contain:

1. Line 1 contains N1, followed by N1 integers of the first array

2. Line 2 contains N2, followed by N2 integers of the second array

Output Format

The intersection of the arrays in a single line

Example

Input:

1

3 10 17 57

6 2 7 10 15 57 246

Output:

10 57

Input:

1

6 1 2 3 4 5 6

2 1 6

Output:

1 6

For example:

Input	Result
1 3 10 17 57 6 2 7 10 15 57 246	10 57

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  int main()
3  {
4      int a;
5      scanf("%d",&a);
6      while(a>0)
7      {
8          int m,n;
9          scanf("%d",&n);
10         int arr1[n];
11         for(int i=0;i<n;i++)
12         {
13             scanf("%d",&arr1[i]);
14         }
15         scanf("%d",&m);
16         int arr2[m];
17         for(int i=0;i<m;i++)
18         {
19             scanf("%d",&arr2[i]);
20         }
21         for(int i=0;i<n;i++)
22         {
```

```
23          for(int j=0;j<m;j++)
24          {
25              if(arr1[i]==arr2[j])
26              {
27                  printf("%d ",arr1[i]);
28              }
29          }
30      }
31      a--;
32  }
33 }
```

	Input	Expected	Got	
✔	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	✔
✔	1 6 1 2 3 4 5 2 1 6	1 6	1 6	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

◄ 2-Finding Duplicates-O(n) Time Complexity,O(1) Space Complexity

Jump to...

4- Print Intersection of 2 sorted arrays-O(m+n)Time Complexity,O(1) Space Complexity ►

Started on	Sunday, 10 November 2024, 1:23 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:26 PM
Time taken	3 mins 33 secs
Marks	1.00/1.00
Grade	**30.00** out of 30.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

Input Format

·   The first line contains T, the number of test cases. Following T lines contain:

1.   Line 1 contains N1, followed by N1 integers of the first array

2.   Line 2 contains N2, followed by N2 integers of the second array

Output Format

The intersection of the arrays in a single line

Example

Input:

1

3 10 17 57

6 2 7 10 15 57 246

Output:

10 57

Input:

1

6 1 2 3 4 5 6

2 1 6

Output:

1 6

For example:

Input	Result
1 3 10 17 57 6 2 7 10 15 57 246	10 57

Answer: (penalty regime: 0 %)

```
1   #include<stdio.h>
2   int main()
3 ▾ {
4       int a;
5       scanf("%d",&a);
6       while(a>0)
7 ▾     {
8           int n,m;
9           scanf("%d",&n);
10          int arr1[n];
11          for(int i=0;i<n;i++)
12 ▾         {
13              scanf("%d",&arr1[i]);
14          }
15          scanf("%d",&m);
16          int arr2[m];
17          for(int i=0;i<m;i++)
18 ▾         {
19              scanf("%d",&arr2[i]);
20          }
21          for(int i=0;i<n;i++)
22 ▾         {
```

```
23          for(int j=0;j<m;j++)
24          {
25              if(arr1[i]==arr2[j])
26              {
27                  printf("%d ",arr1[i]);
28              }
29          }
30      }
31      a--;
32  }
33 }
```

	Input	Expected	Got	
✔	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	✔
✔	1 6 1 2 3 4 5 2 1 6	1 6	1 6	✔

Passed all tests! ✔

Correct
Marks for this submission: 1.00/1.00.

◄ 3-Print Intersection of 2 sorted arrays-O(m*n)Time Complexity,O(1) Space Complexity

Jump to...

5- Pair with Difference-O(n^2)Time Complexity,O(1) Space Complexity ►

Started on	Sunday, 10 November 2024, 1:26 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:31 PM
Time taken	4 mins 21 secs
Marks	1.00/1.00
Grade	**4.00** out of 4.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[j] - A[i] = k, i != j.

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as 5 - 1 = 4

So Return 1.

For example:

Input	Result
3 1 3 5 4	1

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,k,flag=0;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    scanf("%d",&k);
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            if(arr[j]-arr[i]==k && i!=j)
            {
                flag=1;
                break;
            }
        }
    }
    if(flag)
    {
        printf("1");
    }
    else
    {
        printf("0");
    }
}
```

	Input	Expected	Got	
✔	3 1 3 5 4	1	1	✔
✔	10 1 4 6 8 12 14 15 20 21 25 1	1	1	✔
✔	10 1 2 3 5 11 14 16 24 28 29 0	0	0	✔
✔	10 0 2 3 7 13 14 15 20 24 25 10	1	1	✔

Passed all tests! ✔

[Correct]
Marks for this submission: 1.00/1.00.

◄ 4-Print Intersection of 2 sorted arrays-O(m+n)Time Complexity,O(1) Space Complexity

Jump to...

6- Pair with Difference -O(n) Time Complexity,O(1) Space Complexity ►

Started on	Sunday, 10 November 2024, 1:29 PM
State	Finished
Completed on	Sunday, 10 November 2024, 1:32 PM
Time taken	2 mins 49 secs
Marks	1.00/1.00
Grade	**4.00** out of 4.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[j] - A[i] = k, i != j.

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as 5 - 1 = 4

So Return 1.

For example:

Input	Result
3 1 3 5 4	1

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,k,flag=0;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    scanf("%d",&k);
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            if(arr[j]-arr[i]==k && i!=j)
            {
                flag=1;
                break;
            }
        }
    }
    if(flag)
    {
        printf("1");
    }
    else
    {
        printf("0");
    }
}
```

	Input	Expected	Got	
✔	3 1 3 5 4	1	1	✔
✔	10 1 4 6 8 12 14 15 20 21 25 1	1	1	✔
✔	10 1 2 3 5 11 14 16 24 28 29 0	0	0	✔
✔	10 0 2 3 7 13 14 15 20 24 25 10	1	1	✔

Passed all tests! ✔

[Correct]
Marks for this submission: 1.00/1.00.

◄ 5-Pair with Difference-O(n^2)Time Complexity,O(1) Space Complexity

Jump to...