RAJALAKSHMIENGINEERINGCOLLEGE

RAJALAKSHMI NAGAR, THANDALAM – 602 105

RAJALAKSHMI
ENGINEERING COLLEGE

CS23331-DESIGNANDANALYSISOFALGORITHM

LABORATORYLABMANUAL

Name : GK GOKUL PRASATH

Year / Branch / Section : II YEAR / AIML / A

Register No. : 231501050

Semester : III SEMESTER

Academic Year : 2024-2025

INDEX

REG.NO:231501050 **NAME: GK GOKUL PRASATH**

YEAR:IIYEAR **BRANCH:AIML** **SEC:A**

S. NO.	DATE	TITLE	PAGE NO.	TEACHER'SSIGNATURE/REMARKS
WEEK01–BASICCPROGRAMS				
1.1		SWAPPINGOFTWONUMBERS		
1.2		ELIGIBILITYCRITERIA		
1.3		GROCERYITEMS		
1.4		BABA'SGIVINGPATTERN		
1.5		PUNCTUALITYINCENTIVE		
1.6		DIVISIBILITYFINDER		
1.7		QUOTIENTANDREMAINDER		
1.8		GREATESTOFALLNUMBERS		
1.9		EVENORODD		
1.10		FACTORIALOFANUMBER		
1.11		SUMOFNNATURALNUMBERS		
1.11		FIBONACCISERIES		
1.12		POWEROFINTEGERS		
1.13		PRIMEORNONPRIME		
1.14		REVERSEOFANINTEGER		
WEEK02-FINDINGTIMECOMPLEXITYOFALGORITHMS				
2.1		COUNTERMETHOD-WHILELOOP		
2.2		COUNTERMETHOD-FORLOOP		
2.3		COUNTERMETHOD-FACTORS		
2.4		COUNTERMETHOD-FUNCTION		
2.5		COUNTERMETHOD-REVERSE		

WEEK03–DIVIDEANDCONQUER				
3.1		**NUMBEROFZEROSINANARRAY**		
3.2		**MAJORITYELEMENT**		
3.3		**FINDINGFLOORVALUE**		
3.4		**TWOELEMENTSSUMTOX**		
3.5		**IMPLEMENTATIONOFQUICKSORT**		
WEEK04–GREEDYALGORITHMS				
4.1		**COINPROBLEM**		
4.2		**COOKIESPROBLEM**		
4.3		**BURGERPROBLEM**		
4.4		**ARRAY SUMMAX PROBLEM**		
4.5		**PRODCUTOFARRAYELEMENTS-MIN**		
WEEK05–DYNAMICPROGRAMMING				
5.1		**PLAYING WITHNUMBERS**		
5.2		**PLAYINGWITHCHESSBOARD**		
5.3		**LONGESTCOMMONSUBSEQUENCE**		
5.4		**LONGESTNON-DECREASING SUBSEQUENCE**		
WEEK06–COMPETITIVEPROGRAMMING				
6.1		**FINDING DUPLICATES-O(N^2) TIMECOMPLEXITY,O(1)SPACECOMPLEXITY**		
6.2		**FINDINGDUPLICATES-O(N)TIME COMPLEXITY,O(1)SPACECOMPLEXITY**		
6.3		**PRINT INTERSECTION OF 2 SORTEDARRAYS-O(M*N)TIME COMPLEXITY,O(1)SPACECOMPLEXITY**		
6.4		**PRINT INTERSECTION OF 2 SORTEDARRAYS-O(M+N)TIME COMPLEXITY,O(1)SPACECOMPLEXITY**		
6.5		**PAIRWITHDIFFERENCE-O(N^2)TIME COMPLEXITY,O(1)SPACECOMPLEXITY**		
6.6		**PAIR WITH DIFFERENCE -O(N) TIMECOMPLEXITY,O(1)SPACECOMPLEXITY**		

WEEK01–BASICCPROGRAMS

EXPERIMENTNO: 1.1 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

SWAPPINGOFTWONUMBERS

GIVENTWONUMBERS,WRITEACPROGRAMTOSWAPTHENUMBERS.

FOREXAMPLE

Input	Result
10 20	20 10

PROGRAM

```
#include<stdio.h>in
t main()
{
int
a;intb;
int temp;
scanf("%d %d",&a,&b);
/*swapping the two
numbers*/temp=a;
a=b;b=t
emp;
printf("%d %d",a,b);
}
```

OUTPUT

	Input	Expected	Got	
✔	10 20	20 10	20 10	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

EXPERIMENTNO: 1.2 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

ELIGIBILITYCRITERIA

WRITEACPROGRAMTOFINDTHEELIGIBILITYOFADMISSIONFORAPROFESSIONALCOURSEBASEDONTHEFOLLOWINGCRITERIA:

MARKS IN MATHS >=

65MARKS IN PHYSICS >=

55MARKSINCHEMISTRY>=5

0OR

TOTALINALLTHREESUBJECTS>=180

SAMPLETESTCASES:T

ESTCASE1:

INPUT

706080

OUTPUT

THECANDIDATEISELIGIBLE

TESTCASE2:

INPUT

508080

OUTPUT

THECANDIDATEISELIGIBLE

TEST CASE

3INPUT

506040

OUTPUT

THECANDIDATEISNOTELIGIBLE

PROGRAM

```
#include<stdio.h>in

t main()

{
	int
	mark1;int
	mark2;int
	mark3;intt
	otal;
	scanf
	("%d	%d	%d",&mark1,&mark2,&mark3
	);total=mark1+mark2+mark3;

	if(mark1>=65 && mark2>=55 && mark3>=50 && total>=180)
	{
		printf("The candidate is eligible");
	}
	else if(total>=180)
	{
		printf("The candidate is eligible");
	}
	else{
```

OUTPUT

	Input	Expected	Got	
✔	70 60 80	The candidate is eligible	The candidate is eligible	✔
✔	50 80 80	The candidate is eligible	The candidate is eligible	✔

Passed all tests! ✔

EXPERIMENTNO: 1.3 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

GROCERYITEMS

MALINI GOES TO BESTSAVE HYPER MARKET TO BUY GROCERY ITEMS. BESTSAVEHYPERMARKETPROVIDES10%DISCOUNTONTHEBILLAMOUNTBWHENEVER THEBILLAMOUNTBISMORETHANRS.2000.

THEBILLAMOUNTBISPASSEDASTHEINPUTTOTHEPROGRAM.THEPROGRAMMUSTPRIN TTHEFINALAMOUNTAPAYABLEBYMALINI.

INPUTFORMAT:

THEFIRSTLINEDENOTESTHEVALUEOFB.

OUTPUTFORMAT:

THEFIRSTLINECONTAINSTHEVALUEOFTHEFINALPAYABLEAMOUNTA**.**

EXAMPLEINPUT/OUTPUT1:I

NPUT:

1900

OUTPUT:

1900

EXAMPLEINPUT/OUTPUT2:I

NPUT:

3000

OUTPUT:

2700

PROGRAM

```
#include<stdio.h>in
t main()
{
	int b;

	int discount;
	scanf("%d",&b);if(
	b>2000)
	{
		discount=b*0.10;

		printf("%d",b-discount);
	}
	else
	printf("%d",b);
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 1.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

BABA'SGIVINGPATTERN

BABA IS VERY KIND TO BEGGARS AND EVERY DAY BABA DONATES HALF OF THEAMOUNT HE HAS WHEN EVER A BEGGAR REQUESTS HIM. THE MONEY M LEFT IN BABA'SHAND IS PASSED AS THE INPUT AND THE NUMBER OF BEGGARS B WHO RECEIVED THEALMSAREPASSEDASTHEINPUT.THEPROGRAMMUSTPRINTTHEMONEYBABAHADINTHEBEGINNINGOFTHEDAY.

INPUTFORMAT:

THE FIRST LINE DENOTES THE VALUE OF M.THESECONDLINEDENOTESTHEVALUEOFB.

OUTPUTFORMAT:

THE FIRST LINE DENOTES THE VALUE OF MONEY WITH BABA IN THE BEGINNING OF THEDAY.

EXAMPLEINPUT/OUTPUT:

INPUT:

100
2

OUTPUT:

400

EXPLANATION:

Babadonatedtotwobeggars.Sowhenheencounteredsecondbeggarhehad100*2= Rs.200andwhenheencountered1sthehad200*2=Rs.400.

PROGRAM

```
#include<stdio.h>in
t main()
{
        int
        money;int
        beggar;inta
        mount;
        scanf("%d %d",&money,&beggar);

        amount=money*beggar*
        2;printf("%d",amount);
}
```

OUTPUT

	Input	Expected	Got	
✔	100 2	400	400	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

EXPERIMENTNO: 1.5 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PUNCTUALITYINCENTIVE

THECEOOFCOMPANYABCINCWANTEDTOENCOURAGETHEEMPLOYEESCOMINGON TIME TO THE OFFICE. SO HE ANNOUNCED THAT FOR EVERY CONSECUTIVE DAYAN EMPLOYEE COMES ON TIME IN A WEEK (STARTING FROM MONDAY TOSATURDAY), HE WILL BE AWARDED RS.200 MORE THAN THE PREVIOUS DAY AS"PUNCTUALITY INCENTIVE". THE INCENTIVE I FOR THE STARTING DAY (IE ONMONDAY) IS PASSED AS THE INPUT TO THE PROGRAM. THE NUMBER OF DAYS N ANEMPLOYEE CAME ON TIME CONSECUTIVELY STARTING FROM MONDAY IS ALSOPASSED AS THE INPUT. THE PROGRAM MUST CALCULATE AND PRINT THE"PUNCTUALITYINCENTIVE"POFTHEEMPLOYEE.

INPUTFORMAT:

THE FIRST LINE DENOTES THE VALUE OF I.THESECONDLINEDENOTESTHEVALUEOFN.

OUTPUTFORMAT:

THEFIRSTLINEDENOTESTHEVALUEOFP.

EXAMPLEINPUT/OUTPUT:

INPUT:

500
3

OUTPUT:

2100

EXPLANATION:

ONMONDAYTHEEMPLOYEERECEIVESRS.500,ONTUESDAYRS.700,ONWEDNESDAYRS.90 0

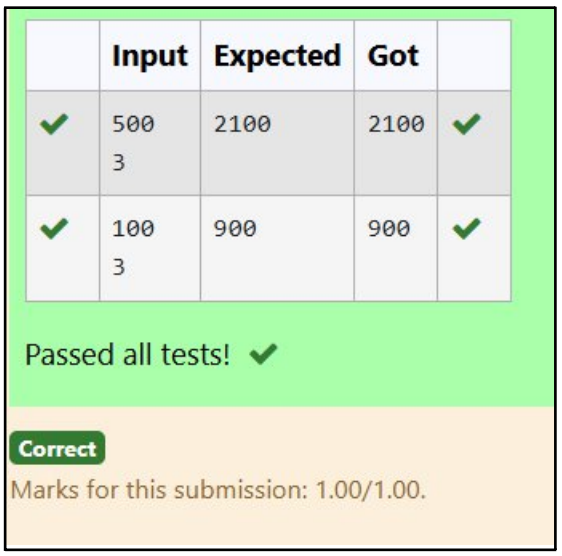
SOTOTAL=RS.2100

PROGRAM

```
#include<stdio.h>in
t main()
{
   int
   a,b,sum=0;scanf("
   %d",&a);
   scanf("%d",&b);
   for(int i=0;i<b;i++)
   {
         sum+=a;
         a=a+200;

   }
   printf("%d",sum);
}
```

OUTPUT



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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

EXPERIMENTNO: 1.6 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

DIVISIBILITYFINDER

TWONUMBERSMANDNAREPASSEDASTHEINPUT.ANUMBERXISALSOPASSEDASTHE INPUT. THE PROGRAM MUST PRINT THE NUMBERS DIVISIBLE BY X FROM N TO M(INCLUSIVEOFMANDN).

INPUTFORMAT:

THE FIRST LINE DENOTES THE VALUE OF
MTHESECONDLINEDENOTESTHEVALUEOFNT
HETHIRDLINEDENOTESTHEVALUEOFX

OUTPUTFORMAT:

NUMBERSDIVISIBLEBYXFROMNTOM,WITHEACHNUMBERSEPARATEDBYASPACE.

BOUNDARYCONDITIONS:

1<=M<=9999999M
< N <=
99999991<=X<=99
99

EXAMPLEINPUT/OUTPUT1:

INPUT:
2
40
7

OUTPUT:
352821147

EXAMPLEINPUT/OUTPUT2:

INPUT:
66
121
11

OUTPUT:

12111099887766

PROGRAM

```
#include<stdio.h>in
t main()
{
	int
	m;int
	n;intx;
	scanf("%d %d",&m,&n);
	scanf("%d",&x);
	for(int i=n;i>m-1;i--)
	{
		if(i%x==0){
			printf("%d ",i);
		}
	}
}
```

OUTPUT

	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.

EXPERIMENTNO: **1.7** **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

QUOTIENT&REMAINDER

WRITEACPROGRAMTOFINDTHEQUOTIENT&REMAINDEROFGIVENINTEGERS

FOREXAMPLE

Input	Result
12 3	4 0

PROGRAM

```
#include<stdio.h>in
t main()
{
   int
   dd;intdr
   ;
   scanf("%d",&dd);
   scanf("%d",&dr);int
   q;
   int
   rem;q=d
   d/dr;
   printf("%d\n",q);re
   m=dd%dr;
```

OUTPUT

	Input	Expected	Got	
✔	12 3	4 0	4 0	✔

Passed all tests! ✔

EXPERIMENTNO: 1.8 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

GREATESTOFALLNUMBERS

WRITEACPROGRAMTOFINDTHEGREATESTNUMBERSOF3INTEGERS.

FOREXAMPLE

Input	Result
10 20 30	30

PROGRAM

```
#include<stdio.h>in
t main()
{
int a;int b;intc;
scanf("%d %d %d",&a,&b,&c);
if(a>b && a>c){
        printf("%d",a);
}
else if(b>c &&
        b>a){printf("%d",b)
        ;
}
else
printf("%d",c);
```

OUTPUT

	Input	Expected	Got	
✔	10 20 30	30	30	✔

Passed all tests! ✔

EXPERIMENTNO: 1.9 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

EVENORODD

WRITEACPROGRAMTOFINDTHENUMBERISODDOREVEN?

FOREXAMPLE

Input	Result
12	Even
11	Odd

PROGRAM

```
#include<stdio.h>in
t main()
{
	int a;
	scanf("%d",&a);

	if(a%2==0){
		printf("Even");
	}
	else
	printf("Odd");
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 1.10 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

FACTORIALOFANUMBER

WRITEAPROGRAMTOFINDTHEFACTORIALOFANUMBER

FOREXAMPLE

Input	Result
5	120

PROGRAM

```
#include<stdio.h>in
t main()
{
	int
	factorial;factoria
	l=1;
	int n;
	scanf("%d",&n);
	for(int i=1;i<=n;i++)
	{
		factorial=factorial*i;
	}
	printf("%d",factorial);
}
```

OUTPUT

	Input	Expected	Got	
✔	5	120	120	✔

Passed all tests! ✔

EXPERIMENTNO: 1.11 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

SUMOFNNATURALNUMBERSWRITEA

CPROGRAMTOFINDTHESUMOFNNATURALNUMBERS**FOREXAMPL**

E

Input	Result
3	6

PROGRAM

```
#include<stdio.h>in
t main(){
	int number;
	scanf("%d",&number);i
	nt i;
	int
	sum;sum
	=0;
	for(i=number;i>=0;i--)
	{
		sum=sum+i;
	}
	printf("%d",sum);
}
```

OUTPUT

	Input	Expected	Got	
✔	3	6	6	✔

Passed all tests! ✔

EXPERIMENTNO: 1.12 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

FIBONACCISERIES

WRITEACPROGRAMTOFINDTHENTHTERMOFFIBONACCISERIES

FOREXAMPLE

Input	Result
0	0
1	1
4	3

PROGRAM

```
#include<stdio.h>in
t main()
{
  int
  a;int
  b;intc;
  int
  sum;b=0;
  c=1;
  sum=0;
  scanf("%d",&a);
  for(int i=0;i<a-
        1;i++){sum=b+c;
        b=c;c=
        sum;
  }
  if(a==1){
        printf("1");
  }else{
        printf("%d",sum);
  }
  }
```

OUTPUT

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

EXPERIMENTNO: 1.13 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

POWEROFINTEGERS

WRITEACPROGRAMTOFINDTHEPOWEROFINTEGERS.

INPUT:

AB

OUTPUT:

A^BVALUE

FOREXAMPLE

Input	Result
2 5	32

PROGRAM

```
#include<stdio.h>#i
nclude<math.h>int
main()
{
	int
	a;intb;
	scanf("%d %d",&a,&b);

	int power;
	power=pow(a,b);
	printf("%d",power);
}
```

OUTPUT

	Input	Expected	Got	
✔	2 5	32	32	✔

Passed all tests! ✔

EXPERIMENTNO: 1.14 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PRIMEORNONPRIME

WRITEACPROGRAMTOFINDWHETHERNUMBERISPRIMEORNOT?

FOREXAMPLE

Input	Result
7	Prime
9	No Prime

PROGRAM

```
#include<stdio.h>
int main()
{
	int number;
	scanf("%d",&number);

	if(number%2==0){
	printf("No Prime");
	}
	else if(number%3==0){
		printf("No Prime");
	}
	else if(number%number==0 &&
		number/number==1){printf("Prime");
	}
	else
	printf("Prime");
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 1.15 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

REVERSEOFANINTEGER

WRITEACPROGRAMTOFINDTHEREVERSEOFANINTEGER.

PROGRAM

```
#include<stdio.h>in
t main()
{
    int n;
    scanf("%d",&n);in
    t reverse;
    reverse=0;i
    nt
    last;last=0;
    while(n!=0){la
    st=n%10;
    reverse=reverse*10+last;n/
    =10;
    }
    printf("%d",reverse);
}
```

OUTPUT

	Input	Expected	Got	
✔	123	321	321	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

WEEK 02 - FINDING TIMECOMPLEXITYOFALGORITHMS

EXPERIMENTNO: 2.1 **DATE:**

REGISTERNO:231501050 **NAME:GK GOKUL PRASATH**

COUNTERMETHOD-WHILELOOP

CONVERTTHEFOLLOWINGALGORITHMINTOAPROGRAMANDFINDITSTIMECOMPLEXITYUSINGTHECOUNTERMETHOD.

```
voidfunction(intn)
{
    int
    i=1;Int
    s=1;
    While(s<=n)
    {
     I++;S
    +=I;
    }
}
```

NOTE:NONEEDOFCOUNTERINCREMENTFORDECLARATIONSANDSCANF()ANDCOUNT VARIABLEPRINTF()STATEMENTS.

INPUT:

APOSITIVEINTEGERN

OUTPUT:

PRINTTHEVALUEOFTHECOUNTERVARIABLE**FOREXAMPLE:**

INPUT	RESULT

9	**12**

PROGRAM

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

OUTPUT

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

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

EXPERIMENTNO: 2.2 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COUNTERMETHOD-FORLOOP

CONVERTTHEFOLLOWINGALGORITHMINTOAPROGRAMANDFINDITSTIMECOMPLEXITYUSINGTHECOUNTERMETHOD.

```
voidfunc(intn)
{
    if(n==1)
    {
       printf("*");
    }
    else
    {
     for(inti=1;i<=n;i++)
     {
        for(intj=1;j<=n;j++)
        {
            printf("*");
            printf("*");br
            eak;
        }
     }
   }
 }
```

NOTE:

NO NEED OF COUNTER INCREMENT FOR DECLARATIONS AND SCANF() AND COUNTVARIABLEPRINTF()STATEMENTS.

INPUT:

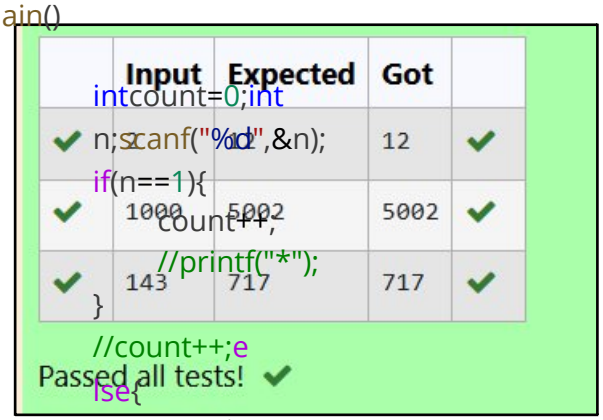
APOSITIVEINTEGERN

OUTPUT:

PRINTTHEVALUEOFTHECOUNTERVARIABLE

PROGRAM

OUTPUT



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

Passed all tests! ✔

```
#include<stdio.h>in
tmain()
{
    intcount=0;int
    n;scanf("%d",&n);
    if(n==1){
        count++;
        //printf("*");
    }
    //count++;e
    lse{
            count++;
            for(inti=1;i<=n;i++)
            {
                count++;
                for(intj=1;j<=n;j++)
                {
                    count++;
                    //printf("*");cou
                    nt++;
                    //printf("*");cou
                    nt++;break;coun
                    t++;
                }
                count++;
            }count++;
        }
        printf("%d",count);
    }
```

EXPERIMENTNO: 2.3 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COUNTERMETHOD-FACTORS

CONVERTTHEFOLLOWINGALGORITHMINTOAPROGRAMANDFINDITSTIMECOMPLEXITYUSINGCOUNTERMETHOD.

```
Factor(num){
{
  for(i=1;i<=num;++i)
  {
   if(num%i==0)
     {
       printf("%d",i);
     }
   }
}
```

NOTE:

NONEEDOFCOUNTERINCREMENTFORDECLARATIONSANDSCANF()ANDCOUNTERVARIABLEPRINTF()STATEMENT.

INPUT:

APOSITIVEINTEGERN

OUTPUT:

PRINTTHEVALUEOFTHECOUNTERVARIABLE

PROGRAM

```
#include<stdio.h>in
t main()
{
	int num;
	scanf("%d",&num);i
	nt count=0;
	int i;
	for(i=1;i<=num;i++)
	{
		count++;
		if(num%i==0)
		{
			count++;
			//printf("%d ",i);
			//count++;
		}count++;
	}count++;
	printf("%d",count);
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 2.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COUNTERMETHOD-FUNCTION

CONVERTTHEFOLLOWINGALGORITHMINTOAPROGRAMANDFINDITSTIMECOMPLEXITYUSINGCOUNTERMETHOD.

```
voidfunction(intn)
{
   intc=0;
   for(int i=n/2; i<n;
      i++)for(intj=1;j<n;j=2*j)
         for(int k=1; k<n; k = k *
            2)c++;
}
```

NOTE:

NO NEED OF COUNTER INCREMENT FOR DECLARATIONS AND SCANF() AND COUNTVARIABLEPRINTF()STATEMENTS.

INPUT:

APOSITIVEINTEGERN

OUTPUT:

PRINTTHEVALUEOFTHECOUNTERVARIABLE

PROGRAM

```
#include<stdio.h>in
t main()
{
    int n;
    scanf("%d",&n);in
    t count=0;
    int
    c=0;coun
    t++;
    for(int
        i=n/2;i<n;i++){count++
        ;
        for(int
            j=1;j<n;j=2*j){count++;
            for(int
                k=1;k<n;k=k*2){count
                ++;
                c++;
                count++;
            }
            count++;
        }
        count++;
    }
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 2.5 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COUNTERMETHOD-REVERSE

CONVERTTHEFOLLOWINGALGORITHMINTOAPROGRAMANDFINDITSTIMECOMPLEXITYUSI NGCOUNTERMETHOD.

```
void reverse(int n)

{
  int rev = 0,

  remainder;while (n !=

  0)

   {
      remainder = n % 10;

      rev = rev * 10 +

      remainder;n/= 10;

   }

print(rev);

}
```

NOTE:

NO NEED OF COUNTER INCREMENT FOR DECLARATIONS AND SCANF() AND COUNTVARIABLEPRINTF()STATEMENTS.

INPUT:

APOSITIVEINTEGERN

OUTPUT:

PRINTTHEVALUEOFTHECOUNTERVARIABLE

PROGRAM

```
#include<stdio.h>in
t main()
{
    int n;
    scanf("%d",&n);in
    t count=0;
    int
    c=0;coun
    t++;
    for(int
        i=n/2;i<n;i++){count++
        ;
        for(int
            j=1;j<n;j=2*j){count++;
            for(int
                k=1;k<n;k=k*2){count
                ++;
                c++;
                count++;
            }
            count++;
        }
        count++;
    }
    count++;
```

OUTPUT

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

Passed all tests! ✔

WEEK03–DIVIDEA NDCONQUER

EXPERIMENTNO: 3.1 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

NUMBEROFZEROSINANARRAY

PROBLEMSTATEMENT

GIVENANARRAYOF1SAND0STHISHASALL1SFIRSTFOLLOWEDBYALL0S.AIMISTO FIND THE NUMBER OF 0S. WRITE A PROGRAM USING DIVIDE AND CONQUER TOCOUNTTHENUMBEROFZEROESINTHEGIVENARRAY.

INPUTFORMAT

FIRSTLINECONTAINSINTEGERM–SIZEOFARRAY

NEXTMLINESCONTAINSMNUMBERS–ELEMENTSOFANARRAY

OUTPUTFORMAT

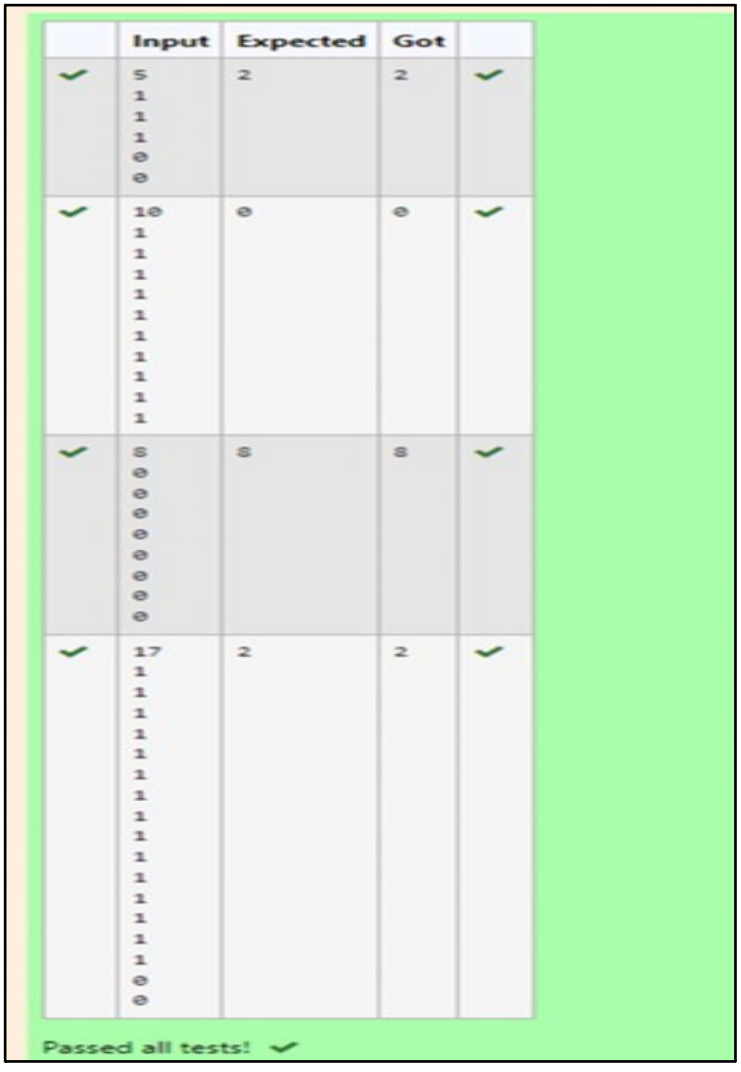
FIRSTLINECONTAINSINTEGER–NUMBEROFZEROESPRESENTINTHEGIVENARRAY.

PROGRAM

```
#include<stdio.h>in
tmain()
{
    int
    n;scanf("%d",&n);
    intarr[n];
    for(int i=0;i<n;i++){scanf("%d",&arr[i]);
    }
    inti;
    int
    count=0;for(i=0;i<
    n;i++)
```

```
    {
        if(arr[i]==0)
        {
                count=count+1;
    }
    }
}
```

OUTPUT



	Input	Expected	Got	
✔	5 1 1 1 0 0	2	2	✔
✔	10 1 1 1 1 1 1 1 1 1 1	0	0	✔
✔	8 0 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 0 0	2	2	✔

Passed all tests! ✔

EXPERIMENTNO: 3.2 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

MAJORITYELEMENT

GIVENANARRAYNUMSOFSIZEN,RETURNTHEMAJORITYELEMENT.

THEMAJORITYELEMENTISTHEELEMENTTHATAPPEARSMORETHAN⌊N/2⌋TIMES.YOUM AYASSUMETHATTHEMAJORITYELEMENTALWAYSEXISTSINTHEARRAY.

EXAMPLE1:

INPUT:NUMS=[3,2,3]

OUTPUT:3

EXAMPLE2:

INPUT:NUMS=[2,2,1,1,1,2,2]

OUTPUT:2

CONSTRAINTS:

N==NUMS.LENGTH1

<=N<=5*104

-231<=NUMS[I]<=231-1

FOREXAMPLE:

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

PROGRAM

```
#include<stdio.h>in
t main(){
    int n;
    scanf("%d",&n);in
    t a[n];
    for(int
        i=0;i<n;i++){scanf("%
        d",&a[i]);
    }
    for(int i=0;i<n;i++){int
        count=0;
        for(int
            j=0;j<n;j++){if(a[i]==
            a[j]){
                count++;
            }
        }
        if(count>n/2){
            printf("%d",a[i]);brea
            k;
        }
    }
```

OUTPUT

	Input	Expected	Got	
✔	3 3 2 3	3	3	✔

Passed all tests! ✔

EXPERIMENTNO: 3.3 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

FINDINGFLOORVALUE

PROBLEMSTATEMENT:

GIVEN A SORTED ARRAY AND A VALUE X, THE FLOOR OF X IS THE LARGESTELEMENTINARRAYSMALLERTHANOREQUALTOX.WRITEDIVIDEANDCONQUERALGORITHMTOFINDFLOOROFX.

INPUTFORMAT

- FIRSTLINECONTAINSINTEGERN–SIZEOFARRAY
- NEXTNLINESCONTAINSNNUMBERS–ELEMENTSOFANARRAY
- LASTLINECONTAINSINTEGERX–VALUEFORX

OUTPUTFORMAT

FIRSTLINECONTAINSINTEGER–FLOORVALUEFORX

PROGRAM

```
#include<stdio.h>in
t main()
{
	int n;
	scanf("%d",&n);in
	t arr[n];
	for(int i=0;i<n;i++)
	{
		scanf("%d",&arr[i]);
	}
	int key=0;
	scanf("%d",&key);int
	floor=arr[0];
	for(int j=1;j<n;j++)
	{
		if(arr[j]>floor && arr[j]<key)
```

```
            floor=arr[j];
        }
        printf("%d",floor);
}
```

OUTPUT

	Input	Expected	Got	
✔	6 1 2 8 10 12 19 5	2	2	✔
✔	5 10 22 85 108 129 100	85	85	✔
✔	7 3 5 7 9 11 13 15 10	9	9	✔

Passed all tests! ✔

EXPERIMENTNO: 3.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

TWOELEMENTSSUMTOX

PROBLEMSTATEMENT:

GIVEN A SORTED ARRAY OF INTEGERS SAY ARR[] AND A NUMBER X. WRITE ARECURSIVEPROGRAMUSINGDIVIDEANDCONQUERSTRATEGYTOCHECKIFTHEREEXIST TWO ELEMENTS IN THE ARRAY WHOSE SUM = X. IF THERE EXIST SUCH TWOELEMENTSTHENRETURNTHENUMBERS,OTHERWISEPRINTAS“NO”.

NOTE:WRITEADIVIDEANDCONQUERSOLUTION

INPUTFORMAT

- FIRSTLINECONTAINSINTEGERN–SIZEOFARRAY
- NEXTNLINESCONTAINSNNUMBERS–ELEMENTSOFANARRAY
- LASTLINECONTAINSINTEGERX–SUMVALUE

OUTPUTFORMAT

- FIRSTLINECONTAINSINTEGER–ELEMENT1
- SECONDLINECONTAINSINTEGER–ELEMENT2(ELEMENT1ANDELEMENTS2TOGETHERSUMSTOVALUE“X”)

PROGRAM

```
#include<stdio.h>in
tmain()
{
    int
    n;scanf("%d",&n);
    intarr[n];

    for(int i=0;i<n;i++){scanf("%d",&arr[i]);
    }

    inti,j;
```

```
        int
        flag;intx;
        scanf("%d",&x);for(i

        =0;i<n;i++){

                for(j=i+1;j<n;j++){if(arr[i]+arr[j]==x){

                                printf("%d\n%d",arr[i],arr[j]);flag=1;

                                break;
                        }

                }
        }
        if(flag==0)printf
        ("No");
}
```

OUTPUT

	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! ✔

EXPERIMENTNO: 3.5 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

IMPLEMENTATIONOFQUICKSORT

WRITEAPROGRAMTOIMPLEMENTTHEQUICKSORTALGORITHM

INPUTFORMAT:

- THEFIRSTLINECONTAINSTHENOOFELEMENTSINTHELIST-N
- THENEXTNLINESCONTAINTHEELEMENTS.

OUTPUT:

SORTEDLISTOFELEMENTS

FOREXAMPLE:

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

PROGRAM

```
#include<stdio.h>in
tmain(){
    int
    n;scanf("%d",&n);i
    ntarr[n];

    for(inti=0;i<n;i++){scanf("%d",&arr[i]);
    }

    for(inti=0;i<n-1;i++){
```

```
            for(intj=0;j<n-i-1;j++)
{
                    if(arr[j]>arr[j+1]){int temp =
                        arr[j];arr[j] =
                        arr[j+1];arr[j+1]=temp;
                    }
                }
        }


        for(int i = 0; i < n;
            i++)printf("%d",arr[i]);
        }

        return0;
}
```

OUTPUT

	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! ✔

WEEK04–GREEDYA LGORITHMS

EXPERIMENTNO: 4.1 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COINPROBLEM

WRITEAPROGRAMTOTAKEVALUEVANDWEWANTTOMAKECHANGEFORVRS,AND WE HAVE INFINITE SUPPLY OF EACH OF THE DENOMINATIONS IN INDIANCURRENCY, 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 NOTESNEEDEDTOMAKETHECHANGE.

INPUTFORMAT:

TAKEANINTEGERFROMSTDIN.

OUTPUTFORMAT:

PRINTTHEINTEGERWHICHISCHANGEOFTHENUMBER.

EXAMPLEINPUT:

64

OUTPUT:

4

EXPLANATON:

WENEEDA50RSNOTEANDA10RSNOTEANDTWO2RUPEECOINS.

PROGRAM

```
#include<stdio.h>in
t main()
{
	int value;
	scanf("%d",&value);

	int currency[]={1000,500,100,50,20,10,5,2,1};
	int totalcurrency;
	totalcurrency=sizeof(currency)/sizeof(currency[0]);int

	count=0;

	for(int i=0;i<totalcurrency;i++)
	{
		if(value==0)
		{
			break;
		}
		count=count+(value/currency[i]);

		value=value%currency[i];
	}
	printf("%d",count);
}
```

OUTPUT

	Input	Expected	Got	
✔	49	5	5	✔

Passed all tests! ✔

EXPERIMENTNO: 4.2 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

COOKIESPROBLEM

ASSUMEYOUAREANAWESOMEPARENTANDWANTTOGIVEYOURCHILDRENSOMECOOKIES.BUT,YOUSHOULDGIVEEACHCHILDATMOSTONECOOKIE.

EACH CHILD I HAS A GREED FACTOR G[I], WHICH IS THE MINIMUM SIZE OF A COOKIETHAT THE CHILD WILL BE CONTENT WITH; AND EACH COOKIE J HAS A SIZE S[J].
IFS[J]>=G[I],WECANASSIGNTHECOOKIEJTOTHECHILDI,ANDTHECHILDIWILLBECONTENT. YOUR GOAL IS TO MAXIMIZE THE NUMBER OF YOUR CONTENT CHILDRENANDOUTPUTTHEMAXIMUMNUMBER.

EXAMPLE1:

INPUT:

3

123

2

11

OUTPUT:

1

EXPLANATION:

- YOUHAVE3CHILDRENAND2COOKIES.THEGREEDFACTORSOF3CHILDRENARE1, 2,3.
- ANDEVENTHOUGHYOUHAVE2COOKIES,SINCETHEIRSIZEISBOTH1,YOUCOULDONLYMAKETHECHILDWHOSEGREEDFACTORIS1CONTENT.
- YOUNEEDTOOUTPUT1.

CONSTRAINTS:

1<=G.LENGTH<=3*10^4

0<=S.LENGTH<=3*10^4

1<=G[I],S[J]<=2^31-1

PROGRAM

```
#include<stdio.h>int
main(){
    int
    n;scanf("%d",&n);
    intgreedfactor[n];
    for (int i = 0; i < n; i++)
        {scanf("%d",&greedfactor[i]);
    }
    intm;scanf("%d",
    &m);intcookiesize[m]
    ;
    for (int j = 0; j < m; j++)
        {scanf("%d",&cookiesize[j]);
    }
    for(inti=0;i<n-1;i++){
        for(intj=0;j<n-i-1;j++){
            if(greedfactor[j]>greedfactor[j+1]){int temp =
                greedfactor[j];greedfactor[j] = greedfactor[j +
                1];greedfactor[j+1]=temp;
            }
        }
    }
    for(inti=0;i<m-1;i++){
        for(intj=0;j<m-i-1;j++){
            if(cookiesize[j]>cookiesize[j+1]){int temp =
                cookiesize[j];cookiesize[j] = cookiesize[j +
                1];cookiesize[j+1]=temp;
            }
        }
    }
    int i =
    0;intj=0;
    intcontents=0;
    while(i<n&&j<m){
        if(cookiesize[j]>=greedfactor[i]){contents++;
            i++;
        }j++;
    }
    printf("%d\n",contents);return0;
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 4.3 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

BURGERPROBLEM

APERSONNEEDSTOEATBURGERS.EACHBURGERCONTAINSACOUNTOFCALORIE.AFTEREATINGTHEBURGER,THEPERSONNEEDSTORUNADISTANCETOBURNOUTHIS CALORIES. IF HE HAS EATEN I BURGERS WITH C CALORIES EACH, THEN HE HASTO RUN AT LEAST 3I * CKILOMETERS TO BURN OUT THE CALORIES. FOREXAMPLE,IF HE ATE 3 BURGERS WITH THE COUNT OF CALORIE IN THE ORDER: [1, 3, 2], THEKILOMETERS HE NEEDS TO RUN ARE (30 * 1) + (31 * 3) + (32 * 2) = 1 + 9 + 18 = 28.BUTTHIS IS NOT THE MINIMUM, SO NEED TO TRY OUT OTHER ORDERS OF CONSUMPTIONAND CHOOSE THE MINIMUM VALUE. DETERMINE THE MINIMUM DISTANCE .HENEEDS TO RUN. NOTE: HE CAN EAT BURGER IN ANY ORDER AND USE AN EFFICIENTSORTINGALGORITHM.APPLYGREEDYAPPROACHTOSOLVETHEPROBLEM.

INPUTFORMAT

- FIRSTLINECONTAINSTHENUMBEROFBURGERS
- SECONDLINECONTAINSCALORIESOFEACHBURGERWHICHISNSPACE-SEPARATEINTEGERS

OUTPUTFORMAT

- PRINT:MINIMUMNUMBEROFKILOMETERSNEEDEDTORUNTOBURNOUTTHE CALORIES

SAMPLEINPUT

3

5107

SAMPLEOUTPUT

76

FOREXAMPLE

Test	Input	Result
Test Case 1	3 1 3 2	18

PROGRAM

```
#include<stdio.h>#i
nclude<math.h>int
main(){
    int
    n=0;scanf("%d",&
    n);inta[n];
    for(int
        i=0;i<n;i++){scanf("%
        d",&a[i]);
    }
    for(int i=0;i<n-1;i++){for(intj=0;j<n-
        i-1;j++){
            if(a[j]>a[j+1]){int
                temp=a[j];a[j]=a[
                j+1];a[j+1]=temp
                ;
            }
        }
    }
    int j=n-
    1;intsum=0;
    for(int
        i=0;i<n;i++){sum=sum+((pow(n
        ,i))*a[j]);j--;
    }
```

OUTPUT

	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! ✔

EXPERIMENTNO: 4.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

ARRAYSUMMAXPROBLEM

GIVENANARRAYOFNINTEGER,WEHAVETOMAXIMIZETHESUMOFARR[I]*I,WHERE I IS THE INDEX OF THE ELEMENT (I = 0, 1, 2, ..., N).WRITE AN ALGORITHMBASEDONGREEDYTECHNIQUEWITHACOMPLEXITYO(NLOGN).

INPUTFORMAT:

- FIRSTLINESPECIFIESTHENUMBEROFELEMENTS-N
- THENEXTNLINESCONTAINTHEARRAYELEMENTS.

OUTPUTFORMAT:

MAXIMUMARRAYSUMTOBEPRINTED.

SAMPLEINPUT:

5

25340

SAMPLEOUTPUT:

40

PROGRAM

```
#include<stdio.h>in
t main(){
	int n;
	scanf("%d",&n);in
	t arr[n];
	for(int i=0;i<n;i++)
	{
		scanf("%d ",&arr[i]);
	}
	for(int i=0;i<n-1;i++)
	{
```

```
            for(int j=0;j<n-i-
                1;j++){if(arr[j]>arr[j+1]){in
                t temp=arr[j];
                arr[j]=arr[j+1];arr[j
                +1]=temp;
            }
        }
}

int maximum=0;
for(int i=0;i<n;i++){
        maximum=maximum+(arr[i]*i);
}printf("%d\n",maximum);
}
```

OUTPUT

	Input	Expected	Got	
✔	5 2 5 3 4 0	40	40	✔
✔	10 2 2 2 4 4 3 3 5 5 5	191	191	✔
✔	2 45 3	45	45	✔

Passed all tests! ✔

EXPERIMENTNO: 4.5 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PRODCUTOFARRAYELEMENTS-MIN

GIVENTWOARRAYSARRAY_ONE[]ANDARRAY_TWO[]OFSAMESIZEN.WENEEDTOFIRST REARRANGE THE ARRAYS SUCH THAT THE SUM OF THE PRODUCT OF PAIRS(1ELEMENTFROMEACH)ISMINIMUM.THATISSUM(A[I]*B[I])FORALLIISMINIMUM.

FOREXAMPLE

Input	Result
3 1 2 3 4 5 6	28

PROGRAM

```
#include
<stdio.h>#include
<stdlib.h>int main() {
    int n;
    scanf("%d", &n);int
    arrayOne[n];intarr
    ayTwo[n];
    for (int i=0;i<n;i++) {
        scanf("%d",&arrayOne[i]);
    }
    for (int i=0;i<n;i++) {
        scanf("%d",&arrayTwo[i]);
    }
    for (int i=0;i<n-1;i++) {
        for (int j=0;j<n-i-1;j++) {
            if (arrayOne[j]>arrayOne[j+1]) {int
                temp = arrayOne[j];
                arrayOne[j]=arrayOne[j +
                1];arrayOne[j+1]=temp;
            }
        }
    }
    for (int i=0;i<n-1;i++) {
        for (int j=0;j<n-i-1;j++) {
            if (arrayTwo[j]<arrayTwo[j+1]) {
```

```
                    int temp=arrayTwo[j];
                    arrayTwo[j]=arrayTwo[j+1];arr
                    ayTwo[j+1]=temp;
                }
            }
        }
        int minimumsum = 0;
        for (int i = 0; i < n; i++) {
            minimumsum=minimumsum+arrayOne[i]*arrayTwo[i];
        }
        printf("%d\n", minimumsum);
}
```

OUTPUT

	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	✔

WEEK – 05PLAYINGWITHNUMBERS

EXPERIMENTNO: 5.1 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PLAYINGWITHNUMBERS

PLAYINGWITHNUMBERS:

RAM AND SITA ARE PLAYING WITH NUMBERS BY GIVING PUZZLES TO EACHOTHER.NOWITWASRAMTERM,SOHEGAVESITAAPOSITIVEINTEGER‘N’ANDTWONUMBERS1AND3.HEASKEDHERTOFINDTHEPOSSIBLEWAYSBYWHICHTHENUMBER N CAN BE REPRESENTED USING 1 AND 3.WRITE ANY EFFICIENTALGORITHMTOFINDTHEPOSSIBLEWAYS.

EXAMPLE1:

INPUT:

6

OUTPUT:

6

EXPLANATION:

THEREARE6WAYSTO6REPRESENTNUMBERWITH1AND31+1+1+
1+1+1
3+3
1+1+1+3
1+1+3+1
1+3+1+1
3+1+1+1

INPUTFORMAT

FIRSTLINECONTAINSTHENUMBERN

OUTPUTFORMAT

PRINT:

THENUMBEROFPOSSIBLEWAYS‘N’CANBEREPRESENTEDUSING1AND3

SAMPLEINPUT

6

SAMPLEOUTPUT

6

PROGRAM

```
#include <stdio.h>int
main() {
        long n;
        scanf("%ld", &n);if
        (n < 0) {
                return 0;
        }
        long array[n +
        1];array[0] = 1;
        array[1] = 1;
        array[2] = 1;
        array[3] = 2;
        for (long i = 4; i <= n; i++) {
                array[i] = array[i - 1] + array[i - 3];
        }
        printf("%ld\n", array[n]);return
        0;
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 5.2 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PLAYINGWITHCHESSBOARD

PLAYINGWITHCHESSBOARD:

RAM IS GIVEN WITH AN N*N CHESSBOARD WITH EACH CELL WITH A MONETARYVALUE. RAM STANDS AT THE (0,0), THAT THE POSITION OF THE TOP LEFT WHITEROOK. HE IS BEEN GIVEN A TASK TO REACH THE BOTTOM RIGHT BLACK ROOKPOSITION (N-1, N-1) CONSTRAINED THAT HE NEEDS TO REACH THE POSITION BYTRAVELINGTHEMAXIMUMMONETARYPATHUNDERTHECONDITIONTHATHECANONLY TRAVEL ONE STEP RIGHT OR ONE STEP DOWN THE BOARD. HELP RAM TOACHIEVEITBYPROVIDINGANEFFICIENTDPALGORITHM.

EXAMPLE:

INPUT

3

124

234

871

OUTPUT:

19

EXPLANATION:

TOTALLYTHEREWILLBE6PATHSAMONGTHATTHEOPTIMALISOPTIMALPA

THVALUE:1+2+8+7+1=19

INPUTFORMAT

- FIRSTLINECONTAINSTHEINTEGERN
- THENEXTNLINESCONTAINTHEN*NCHESSBOARDVALUES

OUTPUTFORMAT

PRINTMAXIMUMMONETARYVALUEOFTHEPATH

PROGRAM

```
#include<stdio.h>
intmaxMonetaryPath(intn,intboard[n][n])
{
	intdp[n][n];
	dp[0][0]=board[0][0];

	for(intj=1;j<n;j++){
		dp[0][j]=dp[0][j-1]+board[0][j];
	}

	for(inti=1;i<n;i++){
		dp[i][0]=dp[i-1][0]+board[i][0];
	}

	for(inti=1;i<n;i++){for(intj=1;j<n;j++){

			dp[i][j]=board[i][j]+(dp[i-1][j]>dp[i][j-1]?dp[i-1][j]:dp[i][j-1]);

		}
	}
	returndp[n-1][n-1];
}

intmain(){
	int
	n;scanf("%d",&n);
	intboard[n][n];

	for(inti=0;i<n;i++){for(intj=0;j<n;j++){

			scanf("%d",&board[i][j]);
		}
	}

	intmaxValue=maxMonetaryPath(n,board);printf("%d\n",
	maxValue);
	return0;
}
```

OUTPUT

	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! ✔

EXPERIMENTNO: 5.3 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

LONGESTCOMMONSUBSEQUENCE

GIVENTWOSTRINGSFINDTHELENGTHOFTHECOMMONLONGESTSUBSEQUENCE(NEED NOTBECONTIGUOUS)BETWEENTHETWO.

EXAMPLE:

S1:GGTABE

S2:TGATASB

S1: A G G T A B

S2: G X T X A Y B

THELENGTHIS4

SOLVINGITUSINGDYNAMICPROGRAMMING

FOREXAMPLE:

Input	Result
aab azb	2

PROGRAM

```
#include
<stdio.h>#include<stri
ng.h>

intlongestCommonSubsequence(char*s1,char*s2){intm=strle
      n(s1);
      int n =

      strlen(s2);intdp[m+1][n+

      1];

      for(inti=0;i<=m;i++){for(intj=0;j<=n;j++){
            if(i==0||j==0){dp[i][j]=0;
            }elseif(s1[i-1]==s2[j-1]){
                  dp[i][j]=dp[i-1][j-1]+1;
            }else{
                  dp[i][j]=(dp[i-1][j]>dp[i][j-1])?dp[i-1][j]:
        dp[i][j-1];
            }
         }
      }

      returndp[m][n];
}

intmain(){
      chars1[100],s2[100];

      fgets(s1,sizeof(s1),stdin);s1[strcspn(s1,"\n")
      ]='\0';

      fgets(s2,sizeof(s2),stdin);s2[strcspn(s2,"\n")
      ]='\0';
      intlength=longestCommonSubsequence(s1,s2);printf("%d\n",l
      ength);

      return0;
}
```

OUTPUT

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

Passed all tests! ✔

EXPERIMENTNO: 5.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

LONGESTNON-DECREASINGSUBSEQUENCE

PROBLEMSTATEMENT:

FINDTHELENGTHOFTHELONGESTNON-DECREASINGSUBSEQUENCEINAGIVENSEQUENCE.

EXAMPLE:

INPUT:

9

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

THESUBSEQUENCEIS[-1,2,2,2,2,3]

OUTPUT:

6

PROGRAM

```
#include<stdio.h>
    intlongseq(intarr[],intn){intdp[n];
            for(inti=0;i<n;i++){dp[i]=
                    1;
            }
            for(inti=1;i<n;i++){for(intj=0;j<i;j++){
                    if(arr[i]>=arr[j]){
                            dp[i]=(dp[i]>dp[j]+1)?dp[i]:dp[j]+1;
                    }
            }
```

```
}
    int
    maximumlength=0;for(i
    nti=0;i<n;i++){
        if(dp[i]>maximumlength){maximumle
            ngth=dp[i];
        }
    }
    returnmaximumlength;
    }
    intmain()
{
    int
    n;scanf("%d",&n);

    intarr[n];
    for(inti=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    intlength=longseq(arr,n);printf("%d\n",leng
    th);

    return0;
}
```

OUTPUT

	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! ✔

WEEK06–COMPETITIVEPROGRAMMING

EXPERIMENTNO: 6.1 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

FINDINGDUPLICATES-O(N^2)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

FINDDUPLICATEINARRAY.

- GIVENAREADONLYARRAYOFNINTEGERSBETWEEN1ANDN,FINDONENUMBERTHATREPEATS.

INPUTFORMAT:

- FIRSTLINE-NUMBEROFELEMENTS
- NLINES-NELEMENTS

OUTPUTFORMAT:

ELEMENTX-THATISREPEATED

FOREXAMPLE:

Input	Result
5 1 1 2 3 4	1

PROGRAM

```
#include<stdio.h>in
tmain()
{
    int
    n,i,count;scanf("
    %d",&n);intarr[n];
```

```
    for(i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++){cou
        nt=0;
        for(int
            j=0;j<n;j++){if(arr[i]==a
            rr[j]){
                count=count+1;
            }
        }
    if(count>1){
}
}
```

OUTPUT

	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! ✔

EXPERIMENTNO: 6.2 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

FINDINGDUPLICATES-O(N)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

FINDDUPLICATEINARRAY.

- GIVENAREADONLYARRAYOFNINTEGERSBETWEEN1ANDN,FINDONENUMBERTHATREPEATS.

INPUTFORMAT:

- FIRSTLINE-NUMBEROFELEMENTS
- NLINES-NELEMENTS

OUTPUTFORMAT:

- ELEMENTX-THATISREPEATED

FOREXAMPLE:

Input	Result
5 1 1 2 3 4	1

PROGRAM

```
#include<stdio.h>in
tmain()
{
    int
    n,i,count;scanf("%
    d",&n);int
    arr[n];for(i=0;i<n;i+
    +)
    {
        scanf("%d",&arr[i]);
    }
```

```
        for(i=0;i<n;i++){coun
                t=0;
                for(int j=0;j<n;j++){if(arr[i]==arr[j]){
                                count=count+1;
                        }
                }
        if(count>1){
                printf("%d\n",arr[i]);break;
        }
}

}
```

OUTPUT

	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! ✔

EXPERIMENTNO: 6.3 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PRINTINTERSECTIONOF2SORTEDARRAYS-O(M*N)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

FINDTHEINTERSECTIONOFTWOSORTEDARRAYSORINOTHERWORDS,

- GIVEN2SORTEDARRAYS,FINDALLTHEELEMENTSWHICHOCCURINBOTHTHE ARRAYS.

INPUTFORMAT

· THEFIRSTLINECONTAINST,THENUMBEROFTESTCASES.FOLLOWINGTLINESCONT AIN:

1. LINE1CONTAINSN1,FOLLOWEDBYN1INTEGERSOFTHEFIRSTARRAY
2. LINE2CONTAINSN2,FOLLOWEDBYN2INTEGERSOFTHESECONDARRAY

OUTPUTFORMAT

- THEINTERSECTIONOFTHEARRAYSINASINGLELINE

EXAMPLE

INPUT:

1

3101757

627101557246

OUTPUT:

1057

INPUT:

1

6123456

216

OUTPUT:

16

FOREXAMPLE:

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

PROGRAM

```
#include<stdio.h>
voidfindIntersection(intarr1[],intv1,intarr2[],intv2){inti=0,j=0;
        while(i<v1&&j<v2){if(arr1[i]==arr2[
                j]){
                        printf("%d",arr1[i]);i++;
                        j++;
                }elseif(arr1[i]<arr2[j]){i++;
                }else{
                        j++;
                }
        }
        printf("\n");
}
intmain(){
        int
        T;scanf("%d",&T);
        while(T--){
                int
                v1;scanf("%d",&v1);i
                ntarr1[v1];
                for(inti=0;i<v1;i++){scanf("%d",&arr1[i]);
                }
                int
                v2;scanf("%d",&v2);i
                ntarr2[v2];
                for(inti=0;i<v2;i++){scanf("%d",&arr2[i]);
                }
                findIntersection(arr1,v1,arr2,v2);
        }
        return0;
}
```

OUTPUT

	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 6 2 1 6	1 6	1 6	✔

Passed all tests! ✔

EXPERIMENTNO: 6.4 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PRINTINTERSECTIONOF2SORTEDARRAYS-
O(M+N)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

FINDTHEINTERSECTIONOFTWOSORTEDARRAYSORINOTHERWORDS,

- GIVEN2SORTEDARRAYS,FINDALLTHEELEMENTSWHICHOCCURINBOTHTHE ARRAYS.

INPUTFORMAT

· THEFIRSTLINECONTAINST,THENUMBEROFTESTCASES.FOLLOWINGTLINESCONT AIN:

1. LINE1CONTAINSN1,FOLLOWEDBYN1INTEGERSOFTHEFIRSTARRAY
2. LINE2CONTAINSN2,FOLLOWEDBYN2INTEGERSOFTHESECONDARRAY

OUTPUTFORMAT

THEINTERSECTIONOFTHEARRAYSINASINGLELINE

EXAMPLE

INPUT:

1

3101757

627101557246

OUTPUT:

1057

INPUT:

1

6123456

216

OUTPUT:

16

FOREXAMPLE:

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

PROGRAM

```
#include <stdio.h>
void findIntersection(int arr1[], int n1, int arr2[], int n2) {int i = 0, j = 0;
	while (i < n1 && j < n2) {
		if (arr1[i] == arr2[j]) {
			printf("%d ",arr1[i]);i++;
			j++;
		} else if (arr1[i] < arr2[j]) {i++;
		} else {
			j++;
		}
	}
	printf("\n");
}
int main() {
	int T;
	scanf("%d", &T);
	while (T--) {
		int n1;
		scanf("%d", &n1);int
		arr1[n1];
		for (int i = 0; i < n1; i++) {scanf("%d",
			&arr1[i]);
		}
		int n2;
		scanf("%d", &n2);int
		arr2[n2];
		for (int i = 0; i < n2; i++) {scanf("%d",
			&arr2[i]);
		}
		findIntersection(arr1, n1, arr2, n2);
	}
	return 0;
}
```

OUTPUT

	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 6 2 1 6	1 6	1 6	✔

Passed all tests! ✔

EXPERIMENTNO: 6.5 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PAIRWITHDIFFERENCE-O(N^2)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

GIVEN AN ARRAY A OF SORTED INTEGERS AND ANOTHER NON NEGATIVEINTEGERK,FINDIFTHEREEXISTS2INDICESIANDJSUCHTHATA[J]-A[I]=K,I!=J.

INPUTFORMAT:

- FIRSTLINEN-NUMBEROFELEMENTSINANARRAY
- NEXTNLINES-NELEMENTSINTHEARRAY
- K-NON-NEGATIVEINTEGER

OUTPUTFORMAT:

- 1-IFPAIREXISTS
- 0-IFNOPAIREXISTS

EXPLANATIONFORTHEGIVENSAMPLETESTCASE:

YESAS5-1=4

SORETURN1.

FOREXAMPLE

Input	Result
3 1 3 5 4	1

PROGRAM

```
#include<stdio.h>in
tmain()
{
```

```
    int
    n;scanf("%d",&n);i
    ntarray[n];
    for(inti=0;i<n;i++)
    {
        scanf("%d",&array[i]);
    }
    int
    d;scanf("%d",&d);i
    ntcount=0;
    for(int
        i=0;i<n;i++){for(intj=0;j<n
        ;j++){
            if(i!=j){
                if(array[j]-array[i]==d){count=count+1;
                }
            }
        }
    }
    if(count==0){
        printf("0");
    }elseprintf("1");
}
```

OUTPUT

	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! ✔

EXPERIMENTNO: 6.6 **DATE:**

REGISTERNO:231501050 **NAME: GK GOKUL PRASATH**

PAIRWITHDIFFERENCE-O(N)TIMECOMPLEXITY,O(1)SPACECOMPLEXITY

GIVENANARRAYAOFSORTEDINTEGERSANDANOTHERNONNEGATIVEINTEGERK,FINDI FTHEREEXISTS2INDICESIANDJSUCHTHATA[J]-A[I]=K,I!=J.

INPUTFORMAT:

- FIRSTLINEN-NUMBEROFELEMENTSINANARRAY
- NEXTNLINES-NELEMENTSINTHEARRAY
- K-NON-NEGATIVEINTEGER

OUTPUTFORMAT

- 1-IFPAIREXISTS
- 0-IFNOPAIREXISTS

EXPLANATIONFORTHEGIVENSAMPLETESTCASE:YESA

S5-1=4

SORETURN1.

FOREXAMPLE

Input	Result
3 1 3 5 4	1

PROGRAM

```
#include<stdio.h>in
tmain()
{
	int
	n;scanf("%d",&n);

	int
	array[n];for(inti=0;i<n;i
	++)
	{
		scanf("%d",&array[i]);
	}
	int
	d;scanf("%d",&d);i
	ntcount=0;

	for(int

		i=0;i<n;i++){for(intj=0;j<n

		;j++){

			if(i!=j){

				if(array[j]-array[i]==d){

					count=count+1;
				}
			}
		}
	}

	if(count==0)
{
		printf("0");
	}
	  else
			printf("1");
}
```

OUTPUT

	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! ✔