BCSE102L - Computer Programming: Python

Digital Footprint

Faculty In charge: JANANI T
Class No.: VL2021220506935

School of Computer Science and Engineering (SCOPE)
Vellore Institute of Technology
Vellore.

BCSE101E - Computer Programming: Python

Table of Contents

Signature of the student (Digital)
Abhinav Dinesh Srivatsa

[Ex. No. M1_CSQ1]

AIM

Write a c program to read a value and check whether it is "even" or "odd". Only positive value can be accepted, otherwise display "Enter only positive number"

Algorithm / Pseudocode

Declare integer Num

Read input and store as Num

If Num > 0, then

  If Num % 2 = 0, then display ‘even’

  Else display ‘odd’

Else display ‘Enter only positive number’

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    int num;
    scanf("%d", &num);
    if (num > 0)
    {
        if (num % 2 == 0)
        {
            printf("even");
        }
        else
        {
            printf("odd");
        }
    }
    else{
        printf("Enter only positive number");
    }
}
```

Output

Test	Input	Result
1	100	even

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

int main()
{
    int num;
    scanf("%d", &num);
    if (num > 0)
    {
        if (num % 2 == 0)
        {
            printf("even");
        }
        else
        {
            printf("odd");
        }
    }
    else{
        printf("Enter only positive number");
    }
}
```

	Test	Input	Expected	Got	
✔	1	100	even	even	✔
✔	2	-1	Enter only positive number	Enter only positive number	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Show all questions on one page

Finish review

[Ex. No. M1_CSQ2]

AIM

Given ‘n’ letters find how many isogram words can be formed? A word is said to be isogram if it is formed without repeating a letter. For example, the word ‘isogram’ itself has the property and ‘Apple’ do not have the property as ‘p’ is repeated in the word

Algorithm / Pseudocode

Declare integer N

Read input and assign to N

Declare integer Fact as 1

Loop from 1 to N as X

    Calculate Fact * X and assign to Fact

Display Fact

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);
    int fact = 1;
    for(int x = 1; x <= n; x++){
        fact *= x;
    }
    printf("%d", fact);
}
```

Output

Test	Input	Result
1	3	6

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);
    int fact = 1;
    for(int x = 1; x <= n; x++){
        fact *= x;
    }
    printf("%d", fact);
}
```

	Test	Input	Expected	Got	
✔	1	3	6	6	✔
✔	2	5	120	120	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Show all questions on one page

Finish review

[Ex. No. M1_CSQ3]

AIM

Each ship serial number begins with a letter indicating the class of the ship. Write a program that reads a ship's first character of serial number and displays the class of the ship.

Class ID	Ship Class
B or b	Battleship
C or c	Cruiser
D or d	Destroyer
F or f	Frigate

Algorithm / Pseudocode

Declare character C

Read input and store as C

Declare integer I as C

If I >= 97, then calculate I – 32 and assign to I

Assign C as I

If C = 'B', then display 'Battleship'

If C = 'C', then display 'Cruiser

If C = 'D', then display 'Destroyer'

If C = 'F', then display 'Frigate'

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    char c;
```

```
        scanf("%c", &c);
        int i = c;
        if (i >= 97)
        {
            i -= 32;
        }
        c = i;
        if (c == 'B')
        {
            printf("Battleship");
        }
        if (c == 'C')
        {
            printf("Cruiser");
        }
        if (c == 'D')
        {
            printf("Destroyer");
        }
        if (c == 'F')
        {
            printf("Frigate");
        }
    }
```

Output

C or c	Cruiser
D or d	Destroyer
F or f	Frigate

For example:

Test	Input	Result
1	B	Battleship

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

int main()
{
    char c;
    scanf("%c", &c);
    int i = c;
    if (i >= 97)
    {
        i -= 32;
    }
    c = i;
    if (c == 'B')
    {
        printf("Battleship");
    }
    if (c == 'C')
    {
        printf("Cruiser");
    }
    if (c == 'D')
    {
```

	Test	Input	Expected	Got	
✔	1	B	Battleship	Battleship	✔
✔	2	c	Cruiser	Cruiser	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Show all questions on one page

Finish review

[Ex. No. M1_CSQ4]

AIM

The table below shows the normal boiling points of several substances. Write a program that prompts the user for the observed boiling point of a substance in °C and identifies the substance if the observed boiling point is within 5% of the expected boiling point. If the data input is more than 5% higher or lower than any of the boiling points in the table, the program should output the message Substance unknown.

Substance	Normal boiling point (°C)
Water	100
Mercury	357
Copper	1187
Silver	2193
Gold	2660

Algorithm / Pseudocode

Declare integer array Temps as array of given temperatures

Declare integer T

Read input and store as T

Declare integer flag as 0

Loop from 0 to length of array Temps as X

    Declare integer Lowert as Temps[X] * 0.95

    Declare integer Uppert as Temps[X] * 1.05

    If T > Lowert and T < Uppert, then

        If X = 0, then display 'Water'

        If X = 1, then display 'Mercury

        If X = 2, then display 'Copper

        If X = 3, then display 'Silver'

        If X = 4, then display 'Gold'

Assign Flag as 1

Break the loop

If Flag = 0, then print 'Substance unknown'

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    int temps[] = {100, 357, 1187, 2193, 2660};
    int t;
    scanf("%d", &t);
    int flag = 0;
    for (int x = 0; x < sizeof(temps) / sizeof(temps[0]); x++)
    {
        int lowert = temps[x] * 0.95;
        int uppert = temps[x] * 1.05;
        if (t > lowert && t < uppert)
        {
            switch (x)
            {
            case 0:
                printf("Water");
                break;
            case 1:
                printf("Mercury");
                break;
            case 2:
                printf("Copper");
                break;
            case 3:
                printf("Silver");
                break;
            case 4:
                printf("Gold");
            }
            flag = 1;
            break;
        }
    }
    if (flag == 0)
```

```
        {
            printf("Substance unknown");
        }
    }
```

Output

Substance	Normal boiling point (°C)
Water	100
Mercury	357
Copper	1187
Silver	2193
Gold	2660

For example:

Test	Input	Result
1	355	Mercury

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main()
{
    int temps[] = {100, 357, 1187, 2193, 2660};
    int t;
    scanf("%d", &t);
    int flag = 0;
    for (int x = 0; x < sizeof(temps) / sizeof(temps[0]); x++)
    {
        int lowert = temps[x] * 0.95;
        int uppert = temps[x] * 1.05;
        if (t > lowert && t < uppert)
        {
            switch (x)
            {
            case 0:
                printf("Water");
                break;
            case 1:
                printf("Mercury");
                break;
```

	Test	Input	Expected	Got	
✔	1	355	Mercury	Mercury	✔
✔	2	2663	Gold	Gold	✔
✔	4	80	Substance unknown	Substance unknown	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Show all questions on one page

Finish review

[Ex. No. M1_CSQ5]

AIM

Check whether a number is prime or not using while statement, otherwise print "Not Prime"

Algorithm / Pseudocode

Declare integer N

Declare integer flag as 0

Read input and assign to N

Declare integer X as 2

Loop while X < N / 2

    If N % X = 0, then display 'Not Prime', set Flag as 1 and break loop

    Increment X by 1

If Flag is 0, then display 'Prime'

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    int n;
    int flag = 0;
    scanf("%d", &n);
    int x = 2;
    while (x < n / 2)
    {
        if (n % x == 0)
        {
            printf("Not Prime");
            flag = 1;
            break;
        }
        x++;
    }
```

```
if (flag == 0)
{
    printf("Prime");
}
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4

Finish attempt ...

Answer: (penalty regime: 0 %)

```
//21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer N
Declare integer flag as 0
Read input and assign to N
Declare integer X as 2
Loop while X < N / 2
    If N % X = 0, then display 'Not Prime', set Flag as 1 and break
    Increment X by 1
If Flag is 0, then display 'Prime'
*/

#include <stdio.h>

int main()
{
    int n;
    int flag = 0;
    scanf("%d", &n);
    int x = 2;
    while (x < n / 2)
```

Check

	Test	Input	Expected	Got	
✔	1	5	Prime	Prime	✔
✔	2	11	Prime	Prime	✔
✔	4	85	Not Prime	Not Prime	✔

Passed all tests! ✔

[Ex. No. M1_CSQ6]

AIM

Develop a c Program to deal with N loans. Use math.h library for pow(a,b) and printf("%.2f",a) to print a value with two precision.

Write a program to help you figure out what your monthly payment will be, given the car's purchase price, down payment, the monthly interest rate, and the time period over which you will pay back the loan. The formula for calculating your payment is

payment = $iP / (1 - (1 + i)^{-n})$

where P = principal (the amount you borrow)

i = monthly interest rate (1/12 of the annual rate)

n = total number of payments

Total number of payments is usually 36, 48, or 60 (months). Program should then display the amount borrowed and the monthly payment rounded to two decimal places.

Algorithm / Pseudocode

Declare integer N

Read input and assign to N

Declare integers P, D, I, T

Loop for integer X as 0 while less than N

    Read 4 inputs and assign them to P, D, I, T respectively

    If T = 36, 48 or 60, then

        Declare float Loan as P – D

Assign Loan as Loan * I / 1200

Declare float Den as $1 - (1 + I/1200)^{-T}$

Assign Loan as Loan / Den

Display Loan with 2 decimal points

Program Code

```c
//21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <math.h>

int main()
{
    int n;
    scanf("%d", &n);
    int p, d, i, t;
    for (int x = 0; x < n; x++)
    {
        scanf("%d %d %d %d", &p, &d, &i, &t);
        if (t == 36 || t == 48 || t == 60)
        {
            float loan = p - d;
            loan *= (float)i / 1200;
            float den = 1 - pow(1 + (float)i/1200, -(float)t);
            loan /= den;
            printf("%.2f\n", floor(loan * 100) / 100);
        }
    }
}
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

Answer: (penalty regime: 0 %)

```
//21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer N
Read input and assign to N
Declare integers P, D, I, T
Loop for integer X as 0 while less than N
    Read 4 inputs and assign them to P, D, I, T respectively
    If T = 36, 48 or 60, then
        Declare float Loan as P - D
        Assign Loan as Loan * I / 1200
        Declare float Den as 1 - (1 + I/1200)-T
        Assign Loan as Loan / Den
        Display Loan with 2 decimal points
*/

#include <stdio.h>
#include <math.h>

int main()
{
    int n;
    scanf("%d", &n);
```

Check

	Test	Input	Expected	Got	
✔	1	2 400000 100000 10 36 500000 250000 10 48	9680.16 6340.65	9680.16 6340.65	✔
✔	2	3 400000 100000 10 36 500000 250000 10 48 400000 200000 10 60	9680.16 6340.65 4249.41	9680.16 6340.65 4249.41	✔

Passed all tests! ✔

[Ex. No. M2_CSQ1]

AIM

Each year the Department of Traffic Accidents receives accident count reports from several cities and towns across the country. Given details of 'n' days, develop an algorithm and write a program to determine the average number of accidents and for each day, print the difference between the number of accidents on that day and average. For example, if the number of accidents is 5 and the values are 10, 12, 15, 13, 5 then average is 11 and the difference of values are 1, 1, 4, 2, 6.

Algorithm / Pseudocode

Declare integer N

Read input and assign to N

Declare integer array Acc of length N

Declare integer Sum as 0

Loop from 0 to N as X

    Read input and assign to Acc[N]

    Calculate Sum + Acc[N] and assign to Sum

Declare integer Mean as Sum / N

Loop from 0 to N as X

    Calculate absolute value of Acc[N] - Mean and assign to Acc[N]

    Display Acc[N]

Program Code

```c
//21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int n;
    scanf("%d", &n);
```

```
    int acc[n];
    int sum = 0;
    for (int x = 0; x < n; x++)
    {
        scanf("%d", &acc[x]);
        sum += acc[x];
    }
    int mean = sum / n;
    for (int x = 0; x < n; x++)
    {
        acc[x] = abs(mean - acc[x]);
        printf("%d\n", acc[x]);
    }
}
```

Output

difference of values are 1, 1, 4, 2, 6

For example:

Test	Input	Result
1	5 10 12 15 13 5	1 1 4 2 6

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Finish attempt ...

Answer: (penalty regime: 0 %)

```
1  //21BDS0340 Abhinav Dinesh Srivatsa
2  /*
3  Declare integer N
4  Read input and assign to N
5  Declare integer array Acc of length N
6  Declare integer Sum as 0
7  Loop from 0 to N as X
8      Read input and assign to Acc[N]
9      Calculate Sum + Acc[N] and assign to Sum
10 Declare integer Mean as Sum / N
11 Loop from 0 to N as X
12     Calculate absolute value of Acc[N] - Mean and assign to Acc[N]
13     Display Acc[N]
14 */
15
16 #include <stdio.h>
17 #include <stdlib.h>
18
19 int main()
20 {
21     int n;
22
```

Check

	Test	Input	Expected	Got	
✔	1	5 10 12 15 13 5	1 1 4 2 6	1 1 4 2 6	✔
✔	2	4 12 14 16 18	3 1 1 3	3 1 1 3	✔

Passed all tests! ✔

[Ex. No. M2_CSQ2]

AIM

Huffman code is a particular type of optimal prefix code for characters. It is commonly used for lossless data compression. It is a variable-length code derived from frequency of occurrence. Given a string develop an algorithm and write a C program to determine frequency of occurrence of each character in the string.

Algorithm / Pseudocode

Declare character array as Sen

Read input till new line and assign to Sen

Declare integer array Count

Declare integer X as 0

Loop while X < 26

    Assign Count[X] as 0 and increment X

Declare integer Ord

Loop while Sen[X] is not '\0'

    Calculate Ord as integer casted Sen[X]

    If Ord is between 'A' and 'Z', then make it lower case

    If Ord is not space, then increment Count[Ord – 'a']

Loop from 0 to 26 as X

    If Count[X] is not 0, then display the character spaced with the Count[X]

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    char sen[50];
    scanf("%[^\n]s", sen);
```

```
    int count[26];
    int x = 0;
    while (x < 26)
    {
        count[x++] = 0;
    }
    x = 0;
    int ord;
    while (sen[x] != '\0')
    {
        ord = (int)sen[x++];
        if (ord <= 'Z' && ord >= 'A')
        {
            ord += 32;
        }
        if (ord != ' ')
        {
            count[ord - 'a']++;
        }
    }
    for (x = 0; x < 26; x++)
    {
        if (count[x] != 0)
        {
            printf("%c %d\n", x + 'a', count[x]);
        }
    }
}
```

Output

Test	Input	Result
1	All is Well	a 1 e 1 i 1 l 4 s 1 w 1

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    char sen[50];
    scanf("%[^\n]s", sen);
    int count[26];
    int x = 0;
    while (x < 26)
    {
        count[x++] = 0;
    }
    x = 0;
    int ord;
    while (sen[x] != '\0')
    {
        ord = (int)sen[x++];
        if (ord <= 'Z' && ord >= 'A')
        {
            ord += 32;
```

Check

	Test	Input	Expected	Got	
✔	1	All is Well	a 1 e 1 i 1 l 4 s 1 w 1	a 1 e 1 i 1 l 4 s 1 w 1	✔
✔	2	Be Positive	b 1 e 2 i 2 o 1 p 1 s 1 t 1 v 1	b 1 e 2 i 2 o 1 p 1 s 1 t 1 v 1	✔

Passed all tests! ✔

[Ex. No. M2_CSQ3]

AIM

Create C program to add two matrices.

Algorithm / Pseudocode

Declare integers R and C

Read inputs and assign to R and C

Declare integer array Mat1 and Mat2 of dimensions R and C

Loop from 0 to R as X

    Loop from 0 to C as Y

        Read input and assign to Mat1[X][Y]

Loop from 0 to R as X

    Loop from 0 to C as Y

        Read input and assign to Mat2[X][Y]

Loop from 0 to R as X

    Loop from 0 to C as Y

        Display Mat1[X][Y] + Mat2[X][Y]

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

int main()
{
    int r, c;
    scanf("%d %d", &r, &c);
    int mat1[r][c];
    int mat2[r][c];
    for (int x = 0; x < r; x++)
    {
        for (int y = 0; y < c; y++)
        {
```

```
                scanf("%d", &mat1[x][y]);
            }
        }
        for (int x = 0; x < r; x++)
        {
            for (int y = 0; y < c; y++)
            {
                scanf("%d", &mat2[x][y]);
            }
        }
        for (int x = 0; x < r; x++)
        {
            for (int y = 0; y < c; y++)
            {
                printf("%d\n", mat1[x][y] + mat2[x][y]);
            }
        }
}
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

Answer: (penalty regime: 0 %)

```
1  //21BDS0340 Abhinav Dinesh Srivatsa
2  /*
3  Declare integers R and C
4  Read inputs and assign to R and C
5  Declare integer array Mat1 and Mat2 of dimensions R and C
6  Loop from 0 to R as X
7      Loop from 0 to C as Y
8          Read input and assign to Mat1[X][Y]
9  Loop from 0 to R as X
10     Loop from 0 to C as Y
11         Read input and assign to Mat2[X][Y]
12 Loop from 0 to R as X
13     Loop from 0 to C as Y
14         Display Mat1[X][Y] + Mat2[X][Y]
15 */
16
17 #include <stdio.h>
18
19 int main()
20 {
21     int r, c;
22     scanf("%d %d", &r, &c);
```

Check

	Test	Input	Expected	Got	
✔	1	2 2 1 2 3 4 2 4 6 8	3 6 9 12	3 6 9 12	✔
✔	2	3 3 1 2 3 4 5 6 7 8 9 2 4 6 8 10 12 14 16 18	3 6 9 12 15 18 21 24 27	3 6 9 12 15 18 21 24 27	✔

Passed all tests! ✔

[Ex. No. M2_CSQ4]

AIM

Create C program with a function named swap_twoNumbers, using call by reference.

Algorithm / Pseudocode

Void swap_twoNumbers(int *V1, int *V2)

Declare integer Temp as *V1

Assign *V1 as *V2

Assign *V2 as Temp

Declare integers A, B

Read input and assign to A and B

Call swap_twoNumbers(A, B)

Display A and B

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

void swap_twoNumbers(int *i1, int *i2)
{
    int temp = *i2;
    *i2 = *i1;
    *i1 = temp;
}

int main()
{
    int a, b;
    scanf("%d%d", &a, &b);
    swap_twoNumbers(&a, &b);
    printf("%d\n%d", a, b);
}
```

Output

1	2 3	3 2

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Create C program with a function named swap_twoNumbers, using call
Algorithm / Pseudocode
Void swap_twoNumbers(int Derefvar1, int Derefvar2)
    Declare integer Temp as Derefvar2
    Assign Derefvar2 as Derefvar1
    Assign Derefvar1 as Temp
Declare integers A, B
Read input and assign to A and B
Call swap_twoNumbers(A, B)
Display A and B
*/

#include <stdio.h>

void swap_twoNumbers(int *i1, int *i2)
{
    int temp = *i2;
    *i2 = *i1;
    *i1 = temp;
```

Check

	Test	Input	Expected	Got	
✔	1	2 3	3 2	3 2	✔
✔	2	10 20	20 10	20 10	✔

Passed all tests! ✔

[Ex. No. M2_CSQ5]

AIM

Each year the Department of Traffic Accidents receives accident count reports from several cities and towns across the country. Given details of 'n' days, develop an algorithm and write a program to determine the average number of accidents and for each day, print the difference between the number of accidents on that day and average. For example, if the number of accidents is 5 and the values are 10, 12, 15, 13, 5 then average is 11 and the difference of values are 1, 1, 4, 2, 6

Use the following functions signature

void read_Count(int *, int);

float find_Mean(int *, int);

void print_Diff(int *, int, float);

Algorithm / Pseudocode

Void read_Count(int *Array, int N)

    Loop from 0 to N as X

        Read input and assign to Array[X]

Float find_Mean(int *Array, int N)

    Declare float Sum as 0

    Loop from 0 to N as X

        Calculate Sum as Sum + Array[X]

    Return Sum / N

Void print_Diff(int *Array, int N, float Mean)

    Loop from 0 to N as X

        Display absolute value of Mean – Array[X]

Declare integer N

Read input and assign to N

Declare integer array Acc with N spaces

Call read_Count(Acc, N)

Declare float Mean and assign by calling find_Mean(Acc, N)

Call print_Diff(Acc, N, Mean)

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

void read_Count(int *array, int n)
{
    for (int x = 0; x < n; x++)
    {
        scanf("%d", &array[x]);
    }
}

float find_Mean(int *array, int n)
{
    float sum = 0;
    for (int x = 0; x < n; x++)
    {
        sum += array[x];
    }
    return sum / n;
}

void print_Diff(int *array, int n, float mean)
{
    for (int x = 0; x < n; x++)
    {
        printf("%d\n", abs((int)mean - array[x]));
    }
}

int main()
{
    int n;
    scanf("%d", &n);
    int acc[n];
```

```
        read_Count(acc, n);
        float mean = find_Mean(acc, n);
        print_Diff(acc, n, mean);
    }
```

Output

difference of values are 1, 1, 4, 2, 6

Use the following functions signature

void read_Count(int *, int);

float find_Mean(int *, int);

void print_Diff(int *, int, float);

For example:

Test	Input	Result
1	5 10 12 15 13 5	1 1 4 2 6

Answer: (penalty regime: 0 %)

```
5   |       Read input and assign to Array[X]
6   |Float find_Mean(int *Array, int N)
7   |    Declare float Sum as 0
8   |    Loop from 0 to N as X
9   |        Calculate Sum as Sum + Array[X]
10  |    Return Sum / N
11  |Void print_Diff(int *Array, int N, float Mean)
12  |    Loop from 0 to N as X
13  |        Display absolute value of Mean - Array[X]
14  |Declare integer N
15  |Read input and assign to N
16  |Declare integer array Acc with N spaces
17  |Call read_Count(Acc, N)
18  |Declare float Mean and assign by calling find_Mean(Acc, N)
19  |Call print_Diff(Acc, N, Mean)
20  |*/
21  |
22  |#include <stdio.h>
23  |#include <stdlib.h>
24  |
25  |void read_Count(int *array, int n)
26  |{
```

Check

	Test	Input	Expected	Got	
✔	1	5 10 12 15 13 5	1 1 4 2 6	1 1 4 2 6	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

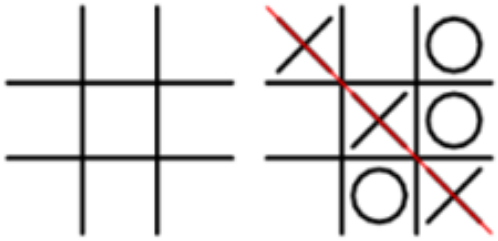
1 2 3 4 5 6

Finish attempt ...

[Ex. No. M2_CSQ6]

AIM

Tic-tac-toe is a paper-and-pencil game for two players, X and O, who take turns marking the spaces n a 3×3 grid. Player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.



Given the board configuration of the tic tac toe game, determine if the board is in either of the following states: empty, player1 wins, player2 wins, draw or intermediate. The board is said to be in initial state if all the cells contain '-1', player1 uses '1' as his coin and player2 uses '2' as his coin. The game is draw when the board is full and no one has won the game. The game is in intermediate state when no one has won and board is not full

Use the following function signatures

void read_Board(int ttt[][3]);

int count_EmptyCell(int ttt[][3]);

int check_Rowwise(int ttt[][3],int);

int check_Colwise(int ttt[][3],int);

int check_Diagonalwise(int ttt[][3],int);

Algorithm / Pseudocode

Void read_Board(int TTT[3][3])

    Loop from 0 to 3 as X

        Loop from 0 to 3 as Y

            Read input and assign to TTT[X][Y]

Int count_EmptyCell(int TTT[3][3])

```
        Declare integer Count as 0
        Loop from 0 to 3 as X
                Loop from 0 to 3 as Y
                        If TTT[X][Y] = -1, then increment Count
        Return Count
Int check_Rowwise(int TTT[3][3])
        Declare integer Val
        Loop from 0 to 3 as X
                Assign Val as TTT[X][0]
                If TTT[X][1] and TTT[X][2] are equal to Val, then return Val
        Return 0
Int check_Colwise(int TTT[3][3])
        Declare integer Val
        Loop from 0 to 3 as X
                Assign Val as TTT[0][X]
                If TTT[1][X] and TTT[2][X] are equal to Val, then return Val
        Return 0
Int check_Diagonalwise(int TTT[3][3])
        Declare integer Val
        Assign Val as TTT[0][0]
        If TTT[1][1] and TTT[2][2] equal Val, then return Val
        Assign Val as TTT[0][2]
        If TTT[1][1] and TTT[2][0] equal Val, then return Val
        Return 0
```

Declare integer array TTT with 3 rows and 3 columns

Call read_Board(TTT)

Declare integer Emptycells and assign it by calling count_EmptyCell(TTT)

If Emptycells = 9, then display 'Empty'

Else if Emptycells = 0, then

Declare integer Rowwin and assign it by calling check_Rowwise(TTT)

Declare integer Colwin and assign it by calling check_Colwise(TTT)

Declare integer Diagwin and assign it by calling check_Diagonalwise(TTT)

If Rowwin, Colwin and Diagwin are 0, then display 'Draw'

Else if Rowwin is not 0, then display 'Player{Rowwin} wins'

Else if Colwin is not 0, then display 'Player{Colwin} wins'

Else if Diagwin is not 0, then display 'Player{Diagwin} wins'

Else display 'Intermediate'

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

void read_Board(int ttt[3][3])
{
    for (int x = 0; x < 3; x++)
    {
        for (int y = 0; y < 3; y++)
        {
            scanf("%d", &ttt[x][y]);
        }
    }
}

int count_EmptyCell(int ttt[3][3])
{
    int count = 0;
    for (int x = 0; x < 3; x++)
```

```
    {
        for (int y = 0; y < 3; y++)
        {
            if (ttt[x][y] == -1)
            {
                count++;
            }
        }
    }
    return count;
}

int check_Rowwise(int ttt[3][3])
{
    int val;
    for (int x = 0; x < 3; x++)
    {
        val = ttt[x][0];
        if (ttt[x][1] == val && ttt[x][2] == val)
        {
            return val;
        }
    }
    return 0;
}

int check_Colwise(int ttt[3][3])
{
    int val;
    for (int x = 0; x < 3; x++)
    {
        val = ttt[0][x];
        if (ttt[1][x] == val && ttt[2][x] == val)
        {
            return val;
        }
    }
    return 0;
}

int check_Diagonalwise(int ttt[3][3])
{
    int val;
    // primary diagonal
    val = ttt[0][0];
```

```
    if (ttt[1][1] == val && ttt[2][2] == val)
    {
        return val;
    }
    // secondary diagonal
    val = ttt[0][2];
    if (ttt[1][1] == val && ttt[2][0] == val)
    {
        return val;
    }
    return 0;
}

int main()
{
    int ttt[3][3];
    read_Board(ttt);
    int emptycells = count_EmptyCell(ttt);
    if (emptycells == 9)
    {
        printf("Empty");
    }
    else if (emptycells == 0)
    {
        int rowwin = check_Rowwise(ttt);
        int colwin = check_Colwise(ttt);
        int diagwin = check_Diagonalwise(ttt);
        if (rowwin == 0 && colwin == 0 && diagwin == 0)
        {
            printf("Draw");
        }
        else if (rowwin != 0)
        {
            printf("Player%d wins", rowwin);
        }
        else if (colwin != 0)
        {
            printf("Player%d wins", colwin);
        }
        else if (diagwin != 0)
        {
            printf("Player%d wins", diagwin);
        }
    }
    else
```

```
        {
            printf("Intermediate");
        }
    }
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

```
2
2
1
-1
```

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Void read_Board(int TTT[3][3])
    Loop from 0 to 3 as X
        Loop from 0 to 3 as Y
            Read input and assign to TTT[X][Y]
Int count_EmptyCell(int TTT[3][3])
    Declare integer Count as 0
    Loop from 0 to 3 as X
        Loop from 0 to 3 as Y
            If TTT[X][Y] = -1, then increment Count
    Return Count
Int check_Rowwise(int TTT[3][3])
    Declare integer Val
    Loop from 0 to 3 as X
        Assign Val as TTT[X][0]
        If TTT[X][1] and TTT[X][2] are equal to Val, then return V
    Return 0
Int check_Colwise(int TTT[3][3])
Declare integer Val
Loop from 0 to 3 as X
```

Check

	Test	Input	Expected	Got	
✔	1	1 2 1 2 1 2 2 1 -1	Intermediate	Intermediate	✔
✔	2	1 2 1 2 1 2 2 1 1	Player1 wins	Player1 wins	✔

Passed all tests! ✔

[Ex. No. M3_CSQ1]

AIM

Design a program in C to find the maximum of given three numbers using pointers.

Algorithm / Pseudocode

Declare integer array Num with 3 spaces

Loop from 0 to 3 as X

  Read input and assign to Num[X]

Declare integer Max as Num[0]

Declare pointer integer Store as Num

Loop from 0 to 3 as X

  If Max < Store[X], then assign Max as Store[X]

Display Max

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int *num = malloc(3 * sizeof(int));
    for (int x = 0; x < 3; x++)
        scanf("%d", num + x);
    int max = *num;
    for (int x = 0; x < 3; x++)
        if (max < *(num + x))
            max = *(num + x);
    free(num);
    printf("%d", max);
}
```

Output

```
20
30
```

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer array Num with 3 spaces
Loop from 0 to 3 as X
    Read input and assign to Num[X]
Declare integer Max as Num[0]
Declare pointer integer Store as Num
Loop from 0 to 3 as X
    If Max < Store[X], then assign Max as Store[X]
Display Max
*/

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int *num = malloc(3 * sizeof(int));
    for (int x = 0; x < 3; x++)
    {
        scanf("%d", num++);
    }
```

Check

	Test	Input	Expected	Got	
✔	1	10 20 30	30	30	✔
✔	2	-5 -3 -2	-2	-2	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

[Ex. No. M3_CSQ2]

AIM

Create a program in C to dynamically allocate integer array. Display the elements of the array using dereferencing operator in the reverse order.

Algorithm / Pseudocode

Declare integer N

Read input and assign to N

Declare integer pointer Arr and assign array of N spaces

Loop from 0 to N as X

  Read input and assign to Arr[X]

Calculate Arr as Arr + N - 1

Loop from N - 1 to -1 as X

  Display value of Arr

  Decrement Arr

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int n;
    scanf("%d", &n);
    int *arr = malloc(n * sizeof(int));
    for (int x = 0; x < n; x++)
        scanf("%d", (arr + x));
    for (int x = n - 1; x > -1; x--)
        printf("%d\n", *(arr + x));
    free(arr);
}
```

Output

1	3 10 20 30	30 20 10

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer N
Read input and assign to N
Declare integer pointer Arr and assign array of N spaces
Loop from 0 to N as X
    Read input and assign to Arr[X]
Calculate Arr as Arr + N - 1
Loop from N - 1 to -1 as X
    Display value of Arr
    Decrement Arr
*/

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int n;
    scanf("%d", &n);
    int *arr = malloc(n * sizeof(int));
    for (int x = 0; x < n; x++)
```

Check

	Test	Input	Expected	Got	
✔	1	3 10 20 30	30 20 10	30 20 10	✔
✔	2	4 -5 -4 -3 -2	-2 -3 -4 -5	-2 -3 -4 -5	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

[Ex. No. M3_CSQ3]

AIM

Consider one dimensional studentsMark array and find maximum and minimum scorer by defining a **function**. Use only pointers, not index to complete the task.

Algorithm / Pseudocode

Int findMin(int *)

    Declare integer Min as Arr's value

    Loop from 0 to 5 as X

        If Min > Arr's value, then assign Min as Arr's value

        Incerement Arr

    Return Min

Int findMax(int *)

    Declare integer Max as Arr's value

    Loop from 0 to 5 as X

        If Max < Arr's value, then assign Max as Arr's value

        Incerement Arr

    Return Max

Declare intger pointer Arr and assign array of 5 spaces

Loop from 0 to 5 as X

    Read input and assign to Arr

    Increment Arr

Calculate Arr as Arr - 5 (Going back to root address)

Display min and max marks by calling findMin(Arr) and findMax(Arr)

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

int findMin_21BDS0340(int *arr)
{
    int min = *arr;
    for (int x = 0; x < 5; x++)
        if (min > *(arr + x))
            min = *(arr + x);
    return min;
}

int findMax_21BDS0340(int *arr)
{
    int max = *arr;
    for (int x = 0; x < 5; x++)
        if (max < *(arr + x))
            max = *(arr + x);
    return max;
}

int main()
{
    int *arr = malloc(5 * sizeof(int));
    for (int x = 0; x < 5; x++)
        scanf("%d", arr + x);
    printf("%d\n%d", findMax_21BDS0340(arr), findMin_21BDS0340(arr));
    free(arr);
}
```

Output

1	70 72 68 94 84	94 68

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Int findMin(int *)
    Declare integer Min as Arr's value
    Loop from 0 to 5 as X
        If Min > Arr's value, then assign Min as Arr's value
        Incerement Arr
    Return Min
Int findMax(int *)
    Declare integer Max as Arr's value
    Loop from 0 to 5 as X
        If Max < Arr's value, then assign Max as Arr's value
        Incerement Arr
    Return Max
Declare intger pointer Arr and assign array of 5 spaces
Loop from 0 to 5 as X
    Read input and assign to Arr
    Increment Arr
Calculate Arr as Arr - 5 (Going back to root address)
Display min and max marks by calling findMin(Arr) and findMax(Arr)
*/
```

Check

	Test	Input	Expected	Got	
✔	1	70 72 68 94 84	94 68	94 68	✔
✔	2	92 70 60 90 80	92 60	92 60	✔

Passed all tests! ✔

[Ex. No. M3_CSQ4]

AIM

The weather station of each city has the detail of rainfall in a year. Given the date and cm of rainfall recorded on that day, write a C program to determine the rainfall recorded in each month of the year and average monthly rainfall in the year.

Algorithm / Pseudocode

Declare integer N

Read input and assign to N

Declare character array Date with 10 spaces

Declare integer array Rain with 12 spaces and assign all indices as 0

Declare integer Cm

Declare character array Month with 3 spaces

Loop from 0 to N

    Read input and assign to Date

    Read input and assign to Cm

    Copy the month portion of Date and assign to Month

    Calculate Rain[Month - 1] as itself + Cm

Declare integer Sum as 0

Loop from 0 to 12 as X

    If Rain[X] is not 0, then display X + 1 and Rain[X]

    Calculate Sum as Sum + Rain[X]

Display Sum / N

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>

int main()
{
    int n;
    scanf("%d", &n);
    char date[10];
    int *rain = calloc(12, sizeof(int));
    int cm;
    char month[3];
    for (int x = 0; x < n; x++)
    {
        scanf("%s", date);
        scanf("%d", &cm);
        strncpy(month, &date[3], 2);
        rain[atoi(month) - 1] += cm;
    }
    int sum = 0;
    for (int x = 0; x < 12; x++)
    {
        if (*(rain + x) != 0)
        {
            printf("%d\n%d\n", (x + 1), *(rain + x));
            sum += *(rain + x);
        }
    }
    printf("%d", (int)round((float)sum / n));
    free(rain);
}
```

Output

Test	Input	Result
1	4 05-01-2022 18 15-01-2022 20 03-02-2022 16 01-03-2022 15	1 38 2 16 3 15 17

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer N
Read input and assign to N
Declare character array Date with 10 spaces
Declare integer array with 12 spaces and assign all indices as 0
Declare integer Cm
Declare character array Month with 3 spaces
Loop from 0 to N
    Read input and assign to Date
    Read input and assign to Cm
    Copy the month portion of Date and assign to Month
    Calculate Rain[Month - 1] as itself + Cm
Declare integer Sum as 0
Loop from 0 to 12 as X
    If Rain[X] is not 0, then display X + 1 and Rain[X]
    Calculate Sum as Sum + Rain[X]
Display Sum / N
*/

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

Check

	Test	Input	Expected	Got	
✔	1	4 05-01-2022 18 15-01-2022 20 03-02-2022 16 01-03-2022 15	1 38 2 16 3 15 17	1 38 2 16 3 15 17	✔
✔	2	5 15-01-2022 16 25-01-2022 20 13-02-2022 15 11-03-2022 15 12-03-2022 18	1 36 2 15 3 33 17	1 36 2 15 3 33 17	✔

Passed all tests! ✔

[Ex. No. M3_CSQ5]

AIM

Design a C program to store Proctee's registered course details. Dynamically allocate memory for n students, scan number of courses, respective course code and credit, dynamically. Print the registration number, registered total credits of the proctees along with common courses.

Algorithm / Pseudocode

Declare integer N, M and Temp

Read input and assign to N

Declare character array pointer Regno as N x 10

Declare integer array pointer Cred as N and assign values as 0

Declare integer Totalcourses as 0

Declare character array pointer Course as Totalcourses x 8

Loop from 0 to N as X

    Read input and store as Regno + X value

    Assign the last index of Regno + X as '\0'

    Read input and store as M

    Assign Course as reallocating Course by expanding the size to M + Totalcourses

    Loop from 0 to M as Y

        Read input and store as Course + Y + Totalcourses value

        Assign the last index of Course + Y + Totalcourses as '\0'

        Read input and assign to Temp

        Calculate Cred + X value as itself + Temp

    Calculate Totalcourses as Totalcourses + M

Loop from 0 to N as X

    Display Regno + X value and Cred + X value

Declare integer array pointer Count with Totalcourses amount of spaces and set all indiecs to 0

Loop from 0 to Totalcourses as X

Loop from X + 1 to Totalcourses as Y

If Course + X value is equal to Course + Y value, then increment Count + X value

Loop from 0 to Totalcourses as X

If Count + X value is N - 1, then display Course + X value

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{
    int n, m, temp;
    scanf("%d", &n);
    char(*regno)[10] = malloc(n * sizeof(*regno));
    int *cred = calloc(n, sizeof(int));
    int total_courses = 0;
    char(*course)[8] = malloc(total_courses * sizeof(*course));
    for (int x = 0; x < n; x++)
    {
        scanf("%s", *(regno + x));
        *(regno + x)[9] = '\0';
        scanf("%d", &m);
        course = realloc(course, (m + total_courses) * sizeof(*course));
        for (int y = 0; y < m; y++)
        {
            scanf("%s", *(course + y + total_courses));
            *(course + y + total_courses)[7] = '\0';
            scanf("%d", &temp);
            *(cred + x) += temp;
        }
        total_courses += m;
    }
    for (int x = 0; x < n; x++)
        printf("%s\n%d\n", *(regno + x), *(cred + x));
```

```
    int *count = calloc(total_courses, sizeof(int));
    for (int x = 0; x < total_courses; x++)
        for (int y = x + 1; y < total_courses; y++)
            if (strcmp(*(course + x), *(course + y)) == 0)
                (*(count + x))++;
    for (int x = 0; x < total_courses; x++)
        if (*(count + x) == (n - 1))
            printf("%s\n", *(course + x));
    free(regno);
    free(cred);
    free(course);
    free(count);
}
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare integer N, M and Temp
Read input and assign to N
Declare character array pointer Regno as N x 10
Declare integer array pointer Cred as N and assign values as 0
Declare integer Totalcourses as 0
Declare character array pointer Course as Totalcourses x 8
Loop from 0 to N as X
    Read input and store as Regno + X value
    Assign the last index of Regno + X as '\0'
    Read input and store as M
    Assign Course as reallocating Course by expanding the size to
    Loop from 0 to M as Y
        Read input and store as Course + Y + Totalcourses value
        Assign the last index of Course + Y + Totalcourses as '\0'
        Read input and assign to Temp
        Calculate Cred + X value as itself + Temp
    Calculate Totalcourses as Totalcourses + M
Loop from 0 to N as X
    Display Regno + X value and Cred + X value
```

Check

	Test	Input	Expected	Got	
✔	1	2 21BCE4001 3 BCSE101 3 BCSE102 4 BCSE103 3 21BCE4002 4 BCSE101 3 BCSE102 4 BCSE103 3 BCSE104 4	21BCE4001 10 21BCE4002 14 BCSE101 BCSE102 BCSE103	21BCE4001 10 21BCE4002 14 BCSE101 BCSE102 BCSE103	✔
✔	2	2 21BCE4003 3 BCSE101 3 BCSE102 4 BCSE103 3 21BCE4004 3 BCSE101 3 BCSE102 4 BCSE104 4	21BCE4003 10 21BCE4004 11 BCSE101 BCSE102	21BCE4003 10 21BCE4004 11 BCSE101 BCSE102	✔

Passed all tests! ✔

[Ex. No. M4_CSQ1]

AIM

Create a structure named student with two attributes, name and cgpa. Define structure variables s1, in the main block (local to main method), s2(global), two structure variable as an array. Scan over all four structure variables and display.

Algorithm / Pseudocode

Declare structure student with 2 fields:

Character array Regis with 9 spaces

Float Cgpa

Declare student S2

Declare student S1 and array S with 2 spaces

Read input and store as S1's Regis and S1's Cgpa

Read input and store as S2's Regis and S2's Cgpa

Loop from 0 to 2 as X

Read input and store as S[x]'s Regis and S1[x]'s Cgpa

Display S1's Regis and S1's Cgpa

Display S2's Regis and S2's Cgpa

Loop from 0 to 2 as X

Display S[x]'s Regis and S1[x]'s Cgpa

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <string.h>

struct student
{
    char regis[9];
```

```
    float cgpa;
} s2;

int main()
{
    struct student s1, s[2];
    scanf("%s%f", s1.regis, &s1.cgpa);
    scanf("%s%f", s2.regis, &s2.cgpa);
    for (int x = 0; x < 2; x++)
    {
        scanf("%s%f", s[x].regis, &s[x].cgpa);
    }
    printf("%s\n%.1f\n", s1.regis, s1.cgpa);
    printf("%s\n%.1f\n", s2.regis, s2.cgpa);
    for (int x = 0; x < 2; x++)
    {
        printf("%s\n%.1f\n", s[x].regis, s[x].cgpa);
    }
}
```

Output

Test	Input	Result
1	21BCE5001 7.1 21BCE5003 7.3 21BCE5004 7.4 21BCE5002 7.2	21BCE5001 7.1 21BCE5003 7.3 21BCE5004 7.4 21BCE5002 7.2

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare structure student with 2 fields:
    Character array Regis with 9 spaces
    Float Cgpa
Declare student S2

Declare student S1 and array S with 2 spaces
Read input and store as S1's Regis and S1's Cgpa
Read input and store as S2's Regis and S2's Cgpa
Loop from 0 to 2 as X
    Read input and store as S[x]'s Regis and S1[x]'s Cgpa
Display S1's Regis and S1's Cgpa
Display S2's Regis and S2's Cgpa
Loop from 0 to 2 as X
    Display S[x]'s Regis and S1[x]'s Cgpa
*/

#include <stdio.h>
#include <string.h>

struct student
```

Check

	Test	Input	Expected	Got	
✔	1	21BCE5001 7.1 21BCE5003 7.3 21BCE5004 7.4 21BCE5002 7.2	21BCE5001 7.1 21BCE5003 7.3 21BCE5004 7.4 21BCE5002 7.2	21BCE5001 7.1 21BCE5003 7.3 21BCE5004 7.4 21BCE5002 7.2	✔

Passed all tests! ✔

[Ex. No. M4_CSQ2]

AIM

Create an employee structure with elements, empid, name, age, dept, designation, salary. Define array of employees and pass to a function to read the values and another function to sort the employees based on age attribute, display only empid.

Algorithm / Pseudocode

Declare structure employee with 6 fields:

Integer Empid

Character array Name with 10 spaces

Integer Age

Character array Dept with 6 spaces

Character array Desig with 15 spaces

Integer Salary

void sort_21BDS0340(struct employee *E)

Read inputs and store as E's Empid, Name, Age, Dept, Desig, Salary

void sort_21BDS0340(struct employee E[3])

Declare employee Temp

Loop from 0 to 2 as X

Loop from 0 to 2 - X as Y

If E[Y]'s Age > E[Y+1]'s Age, then

Assign Temp as E[Y]

Assign E[Y] as E[Y+1]

Assign E[Y+1] as Temp

Declare employee E with 3 spaces

Loop from 0 to 3 as X

  Call getValues_21BDS0340 and pass E[X]

Call sort_21BDS0340 and pass E

Loop from 2 to -1 as X

  Display E[X]'s Empid

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

struct employee
{
    int empid;
    char name[10];
    int age;
    char dept[6];
    char desig[15];
    int salary;
};

void getValues_21BDS0340(struct employee *e)
{
    scanf("%d%s%d%s%s%d", &e->empid, e->name, &e->age, e->dept, e->desig, &e-
>salary);
}

void sort_21BDS0340(struct employee e[3])
{
    struct employee temp;
    for (int x = 0; x < 2; x++)
        for (int y = 0; y < 2 - x; y++)
            if (e[y].age > e[y + 1].age)
            {
                temp = e[y];
                e[y] = e[y + 1];
                e[y + 1] = temp;
```

```
            }
    }

    int main()
    {
        struct employee e[3];
        for (int x = 0; x < 3; x++)
        {
            getValues_21BDS0340(&e[x]);
        }
        sort_21BDS0340(e);
        for (int x = 2; x >= 0; x--)
        {
            printf("%d\n", e[x].empid);
        }
    }
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

```
    Read inputs and store as E's Empid, Name, Age, Dept, Desig, Sa

void sort_21BDS0340(struct employee E[3])
    Declare employee Temp
    Loop from 0 to 2 as X
        Loop from 0 to 2 - X as Y
            If E[Y]'s Age > E[Y+1]'s Age, then
                Assign Temp as E[Y]
                Assign E[Y] as E[Y+1]
                Assign E[Y+1] as Temp
```

Check

	Test	Input	Expected	Got	
✔	1	10001 abc 31 deptA Supervisor 40000 10010 def 28 deptB Trainee 25000 10005 ghi 40 deptA ProdManager 80000	10005 10001 10010	10005 10001 10010	✔
✔	2	10002 abc 25 deptA Supervisor 40000 10006 def 26 deptB Trainee 25000 10008 ghi 27 deptA ProdManager 80000	10008 10006 10002	10008 10006 10002	✔

Passed all tests! ✔

[Ex. No. M4_CSQ3]

AIM

The weather station of each city has the detail of rainfall in a year. Given the date and cm of rainfall recorded on that day, write a C program to determine the rainfall recorded in each month of the year and average monthly rainfall in the year

Note: please use structure for rainfall and date with necessary attributes. Please store date character array appropriately in date structure variable while returning from function.

Algorithm / Pseudocode

Declare structure date with 3 fields:

Integer D

Integer M

Integer Y

Declare structure rain with 2 fields:

Date Date

Integer Cm

struct date dateToStruct(char Array[11])

Declare date D

Assign D.D as the date part of Array

Assign D.M as the month part of Array

Assign D.Y as the year part of Array

Return D

Declare integer N

Read input and assign to N

Declare rain array R with N spaces

Declare character array Datestr with 11 spaces

Loop from 0 to N as X

    Read inputs and assign to Datestr and R[X]'s Cm

    Assign R[X]'s Date by calling dateToStruct and passing Datestr

Declare integer array Month_rain with 12 empty spaces

Declare integer Total_rain as 0

Loop from 0 to N as X

    Calculate Month_rain[R[X]'s Date's D] as itself + R[X]'s Cm

    Calculate Total_rain as itself + R[X]'s Cm

Loop from 0 to N as X

    If Month_rain[X] is not 0, then display X + 1 and Month_rain[X]

Display Total_rain / N

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

struct date
{
    int d;
    int m;
    int y;
};

struct rain
{
    struct date date;
    int cm;
};
```

```c
struct date dateToStruct(char array[11])
{
    struct date d;
    d.d = atoi(array);
    d.m = atoi(&array[3]);
    d.y = atoi(&array[6]);
    return d;
}

int main()
{
    int n;
    scanf("%d", &n);
    struct rain *r = malloc(n * sizeof(struct rain));
    char datestr[11];
    for (int x = 0; x < n; x++)
    {
        scanf("%s%d", datestr, &(r + x)->cm);
        (r + x)->date = dateToStruct(datestr);
    }
    int *month_rain = calloc(12, sizeof(int));
    int total_rain = 0;
    for (int x = 0; x < n; x++)
    {
        month_rain[(r + x)->date.m - 1] += (r + x)->cm;
        total_rain += (r + x)->cm;
    }
    for (int x = 0; x < 12; x++)
        if (month_rain[x] != 0)
            printf("%d\n%d\n", (x + 1), month_rain[x]);
    printf("%d", (int)round((float)total_rain / n));
    free(month_rain);
    free(r);
}
```

Output

For example:

Test	Input	Result
1	4 05-01-2022 18 15-01-2022 20 03-02-2022 16 01-03-2022 15	1 38 2 16 3 15 17

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare structure date with 3 fields:
    Integer D
    Integer M
    Integer Y

Declare structure rain with 2 fields:
    Date Date
    Integer Cm

struct date dateToStruct(char Array[11])
    Declare date D
    Assign D.D as the date part of Array
    Assign D.M as the month part of Array
    Assign D.Y as the year part of Array
    Return D

Declare integer N
Read input and assign to N
Declare rain array R with N spaces
```

Check

	Test	Input	Expected	Got	
✔	1	4 05-01-2022 18 15-01-2022 20 03-02-2022 16 01-03-2022 15	1 38 2 16 3 15 17	1 38 2 16 3 15 17	✔
✔	2	5 15-01-2022 16 25-01-2022 20 13-02-2022 15 11-03-2022 15 12-03-2022 18	1 36 2 15 3 33 17	1 36 2 15 3 33 17	✔

Passed all tests! ✔

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 9 10

Finish attempt ...

[Ex. No. M4_CSQ4]

AIM

Define C program with union definition named ID with three attributes, Aadhar, Pan, VoterId, based on choice from user, assign and access the element.

Algorithm / Pseudocode

Declare union data with 3 fields:

Character array Aadhar with 13 spaces

Character array Pan with 11 spaces

Character array Voter with 11 spaces

Declare union data Id

Declare integer N

Read input and assign to N

Switch N

If 1, then read input and assign to Id's Aadhar and display the same

If 2, then read input and assign to Id's Pan and display the same

If 3, then read input and assign to Id's Voter and display the same

Program Code

```c
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <stdio.h>

union data
{
    char aadhar[13];
    char pan[11];
    char voter[11];
};

int main()
{
```

```
    union data id;
    int n;
    scanf("%d", &n);
    switch (n)
    {
    case 1:
        scanf("%s", id.aadhar);
        printf("%s", id.aadhar);
        break;
    case 2:
        scanf("%s", id.pan);
        printf("%s", id.pan);
        break;
    case 3:
        scanf("%s", id.voter);
        printf("%s", id.voter);
        break;
    }
}
```

Output

1	1 912901237890	912901237890

ABHINAV DINESH SRIVATSA 21BDS0340

1 2 3 4 5 6 7 8 10

Finish attempt ...

Answer: (penalty regime: 0 %)

```
// 21BDS0340 Abhinav Dinesh Srivatsa
/*
Declare union data with 3 fields:
    Character array Aadhar with 13 spaces
    Character array Pan with 11 spaces
    Character array Voter with 11 spaces

Declare union data Id
Declare integer N
Read input and assign to N
Switch N
    If 1, then read input and assign to Id's Aadhar and display th
    If 2, then read input and assign to Id's Pan and display the s
    If 3, then read input and assign to Id's Voter and display the
*/

#include <stdio.h>

union data
{
    char aadhar[13];
```

Check

	Test	Input	Expected	Got	
✔	1	1 912901237890	912901237890	912901237890	✔
✔	2	3 AGEPT1234E	AGEPT1234E	AGEPT1234E	✔

Passed all tests! ✔

[Ex. No. M4_CSQ5]

AIM

Details of employees (emp ID, name, joining date and mobile number) of a company is stored and maintained by the company's IT department. On his birthday, the GM of the company wants to give a surprise gift of Rs.5000 for his employees who joined before 01/01/2010 and whose employee id is divisible by 5. Develop an algorithm and write a C program to display the name of the employees who are eligible to receive the gift and their mobile number

Algorithm / Pseudocode

Declare typedef structure date with 3 fields:

Integer Day

Integer M

Integer Y

Declare typedef structure emp with 4 fields:

Integer Id

Character array Name with 15 spaces

Date Joindate

Character array Mobno with 11 spaces

date splitDate(char Str[10])

Declare Date D

Assign D's Day as the date part of Str

Assign D's Month as the month part of Str

Assign D's Year as the year part of Str

Return D

int isSurprised(emp E)

If E's Id is divisible by 5 and E's Joindate is less than 2010, then return 1

Return 0

Declare integer N

Read input and assign to N

Declare emp array E with N spaces

Declare character array Date with 11 spaces

Loop from 0 to N as X

Read 4 inputs and assign to (E + X)'s Id, (E + X)'s Name, Date and (E + X)'s Mobno

Assign (E + X)'s Joindate by calling splitDate and pass Date as the argument

Loop from 0 to N as X

If isSurprised of *(E + X) is 1, then print (E + X)'s Name and Mobno

Program Code

```c
// 21BDS0340 ABhinav Dinesh Srivatsa

#include <stdio.h>
#include <stdlib.h>

typedef struct
{
    int day;
    int month;
    int year;
} date;

typedef struct
{
    int id;
    char name[15];
    date join_date;
```

```
    char mob_no[11];
} emp;

date splitDate(char str[11])
{
    date d;
    d.day = atoi(&str[0]);
    d.month = atoi(&str[3]);
    d.year = atoi(&str[6]);
    return d;
}

int isSurprised(emp e)
{
    if (e.id % 5 == 0 && e.join_date.year < 2010)
        return 1;
    return 0;
}

int main()
{
    int n;
    scanf("%d", &n);
    emp *e = malloc(n * sizeof(emp));
    char date[11];
    for (int x = 0; x < n; x++)
    {
        scanf("%d%s%s%s", &(e + x)->id, (e + x)->name, date, (e + x)-
>mob_no);
        (e + x)->join_date = splitDate(date);
    }
    for (int x = 0; x < n; x++)
        if (isSurprised(*(e + x)) == 1)
            printf("%s\n%s\n", (e + x)->name, (e + x)->mob_no);
    free(e);
}
```

Output

ABHINAV DINESH SRIVATSA 21BDS0340

```
    Integer M
    Integer Y

Declare typedef structure emp with 4 fields:
    Integer Id
    Character array Name with 15 spaces
    Date Joindate
    Character array Mobno with 11 spaces

date splitDate(char Str[10])
    Declare Date D
    Assign D's Day as the date part of Str
    Assign D's Month as the month part of Str
    Assign D's Year as the year part of Str
    Return D

int isSurprised(emp E)
```

Check

	Test	Input	Expected	Got	
✔	1	3 10001 e1name 10-10-2009 9192939495 10005 e2name 10-05-2008 9192838485 10010 e3name 12-01-2011 9192737475	e2name 9192838485	e2name 9192838485	✔
✔	2	4 10001 e1name 10-10-2009 9192939495 10005 e2name 10-05-2008 9192838485 10010 e3name 12-01-2011 9192737475 10020 e4name 01-10-2005 9192939485	e2name 9192838485 e4name 9192939485	e2name 9192838485 e4name 9192939485	✔

Passed all tests! ✔

[Ex. No. M5_CSQ1]

AIM

In an online examination system, each test will be scheduled for 'x' minutes. The student is free to take up the test on his convenience but once he starts the test, he must complete. Given the start time and the value of 'x' for an examination, develop an algorithm and write a 'C++' code for the examination system to calculate the finish time of the test

Algorithm / Pseudocode

Test:

Declare integer Testtime

Declare integer Hour

Declare integer Minute

Public void readValues():

Declare integer TT, H, M

Assign Testtime as TT

Assign Hour as H

Assign Minute as M

Public void showEndTime():

Calculate Minute as Minute + Testtime

If Minute > 59, then calculate Hour as Hour + Minute / 60 and calculate minute as minute % 60

If Hour > 12, then calculate Hour as Hour - 12

If Hour < 10, then display "0" + Hour + ":"

Else display Hour + ":"

If Minute < 10, then display "0" + Minute

Else display Minute

Declare Test T

Call T.readValues()

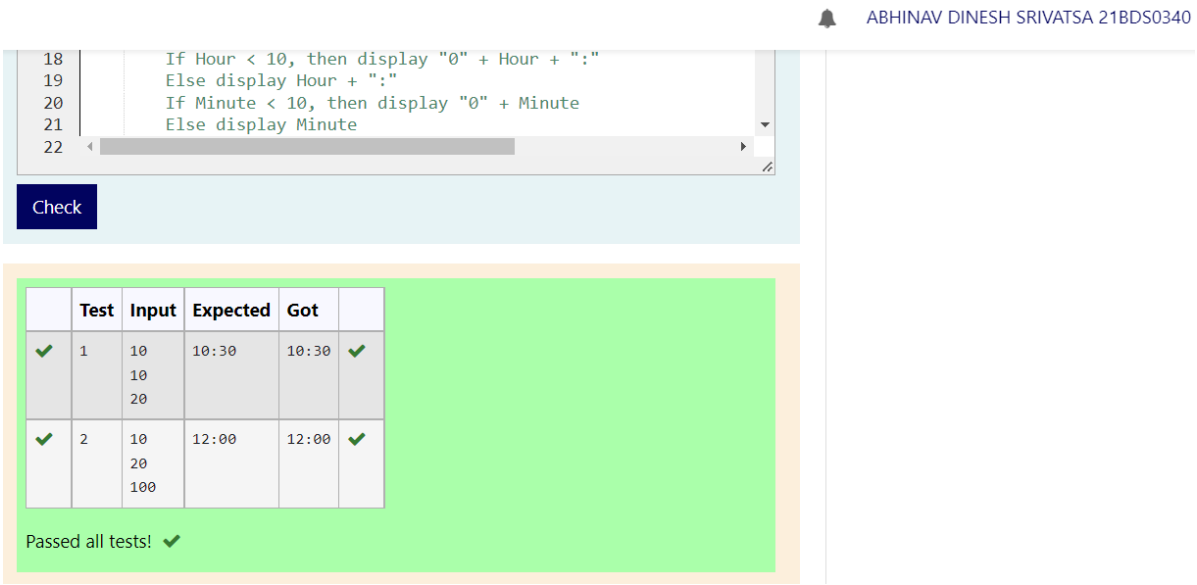
Call T.showEndTIme()

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
using namespace std;

class Test
{
    int testtime;
    int hour;
    int minute;

public:
    void readValues()
    {
        int tt, h, m;
        cin >> h >> m >> tt;
        testtime = tt;
        hour = h;
        minute = m;
    }

    void showEndTime()
    {
        minute += testtime;
        if (minute > 59)
        {
            hour += minute / 60;
            minute %= 60;
        }
        if (hour > 12)
            hour -= 12;

        if (hour < 10)
            cout << "0" << hour << ":";
```

```
            else
                cout << hour << ":";
            if (minute < 10)
                cout << "0" << minute;
            else
                cout << minute;
        }
    };

    int main()
    {
        Test t;
        t.readValues();
        t.showEndTime();
    }
```

Output



[Ex. No. M5_CSQ2]

AIM

Define class rectangle with two attributes length and width along with member functions to setvalues and getvalues. Also define a member function to find the area of rectangle. At least create three objects and print the respective area for all three instances. Use this pointer in your methods.

Algorithm / Pseudocode

Rectangle:

Declare integer Length

Declare integer Width

Public void setLength(int L):

Assign Length as L

Public void setWidth(int W):

Assign Width as W

Public int getLength():

Return Length

Public int getWidth():

Return Width

Public in area(int L, int W):

Return L * W

Declare Rectangle array R with 3 spaces

Declare integer Length and Width

Loop from 0 to 3 as X

Read inputs and assign to Length and Width

Call R[X]'s setLength() and pass Length

Call R[X]'s setWidth() and pass Width

Loop from 0 to 3 as X

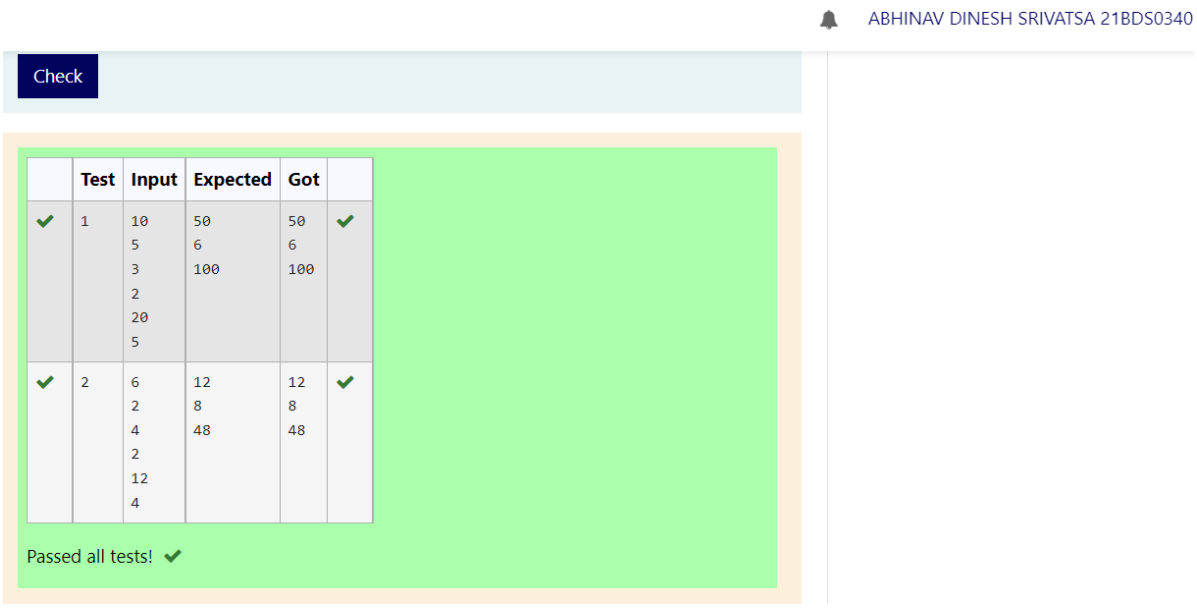
Call R[X]'s area() and pass R[X]'s getLength() and R[X]'s getWidth() and display

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
using namespace std;

class Rectangle
{
    int length;
    int width;

public:
    void setLength(int length)
    {
        this->length = length;
    }

    void setWidth(int width)
    {
        this->width = width;
    }

    int getLength()
    {
        return this->length;
    }

    int getWidth()
    {
```

```
        return this->width;
    }

    int area(int length, int width)
    {
        return length * width;
    }
};

int main()
{
    Rectangle r[3];
    int length, width;
    for (int x = 0; x < 3; x++)
    {
        cin >> length >> width;
        (r + x)->setLength(length);
        (r + x)->setWidth(width);
    }
    for (int x = 0; x < 3; x++)
    {
        cout << (r + x)->area((r + x)->getLength(), (r + x)->getWidth()) <<
"\n";
    }
}
```

Output



[Ex. No. M5_CSQ3]

AIM

Develop C++ program for a company to deal with employee information and create five objects. Find the employee whose designation is Manager and display respective emp id. Also create another method to display designation, by giving emp id.

Algorithm / Pseudocode

Employee:

    Declare public integer Id

    Declare public character array Name with 10 spaces

    Declare public character array Desig with 12 spaces

Void getId(Employee E[5], char Desig[10]):

    Loop from 0 to 5 as X

        If E[X]'s Desig == Desig, then display E[X]'s Id

Void getDesig(Employee E[5], int Id):

    Loop from 0 to 5 as Y

        If E[X]'s Id == Id, then display E[X]'s Desig

Declare Employee array E with 5 spaces

Declare integer Id

Declare character array Name with 10 spaces

Declare character array Desig with 12 spaces

Loop from 0 to 5 as X

    Read inputs and assign to Id, Name and Desig

Assign E[X]'s Id as Id

Assign E[X]'s Name as Name

Assign E[X]'s Desig as Desig

Read input and assign to Desig

Read input and assign to Id

Call getId() and pass E and Desig

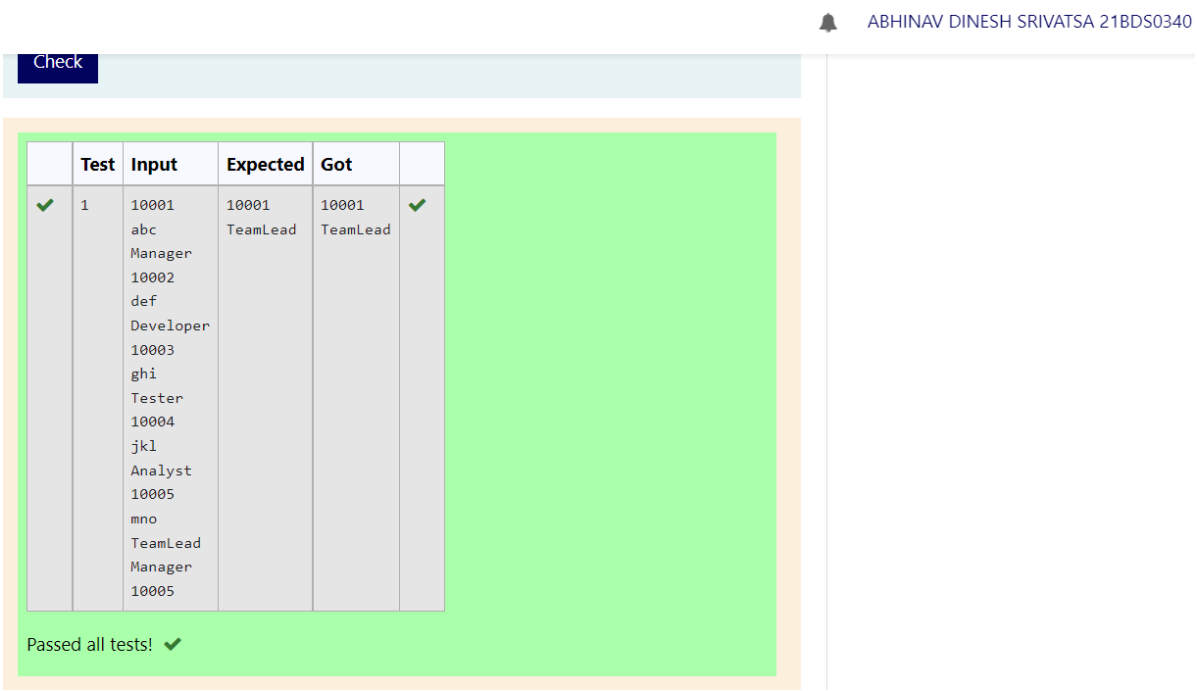
Call getDesig() and pass E and Id

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
#include <string.h>
using namespace std;

class Employee
{
public:
    int id;
    char name[10];
    char desig[12];
};

void getID(Employee e[5], char desig[10])
{
    for (int x = 0; x < 5; x++)
        if (strcmp((e + x)->desig, desig) == 0)
            cout << (e + x)->id << "\n";
}

void getDesig(Employee e[5], int id)
{
    for (int x = 0; x < 5; x++)
        if ((e + x)->id == id)
            cout << (e + x)->desig << "\n";
}

int main()
{
```

```
        Employee e[5];
        int id;
        char name[10];
        char desig[12];
        for (int x = 0; x < 5; x++)
        {
            cin >> id >> name >> desig;
            (e + x)->id = id;
            strcpy((e + x)->name, name);
            strcpy((e + x)->desig, desig);
        }
        cin >> desig;
        cin >> id;
        getID(e, desig);
        getDesig(e, id);
    }
```

Output



[Ex. No. M5_CSQ4]

AIM

Create a class student with necessary attributes specific to each student along with constructors and define few instances. Define few common attributes common to all student with the help of static keyword and declare static member function. Display using a non-static member function, all students' regno.

Algorithm / Pseudocode

Student:

Declare character array Regno with 10 spaces

Declare character array Name with 20 spaces

Declare integer Age

Declare public static character array Sec with 10 spaces

Declare public static character array Uni with 10 spaces

Student(char R[10], char N[20], int A):

Assign Regno as R

ASsign Name as N

Assign Age as A

Void dispRegno():

Display Regno

Declare character array Student's Sec with 10 spaces

Declare character array Student's Uni with 10 spaces

int main():

Read inputs and assign to Student's Sec and Student's Uni

Declare character array Regno with 2 x 10 spaces

Declare character array Name with 2 x 10 spaces

Declare integer array Age with 2 spaces

Loop from 0 to 2 as X

Read inputs and assign to Regno[X], Name[X] and Age[X]

Declare Student array S with 2 spaces and call constructor by passing Regno, Name and Age

Loop from 0 to 2 as X

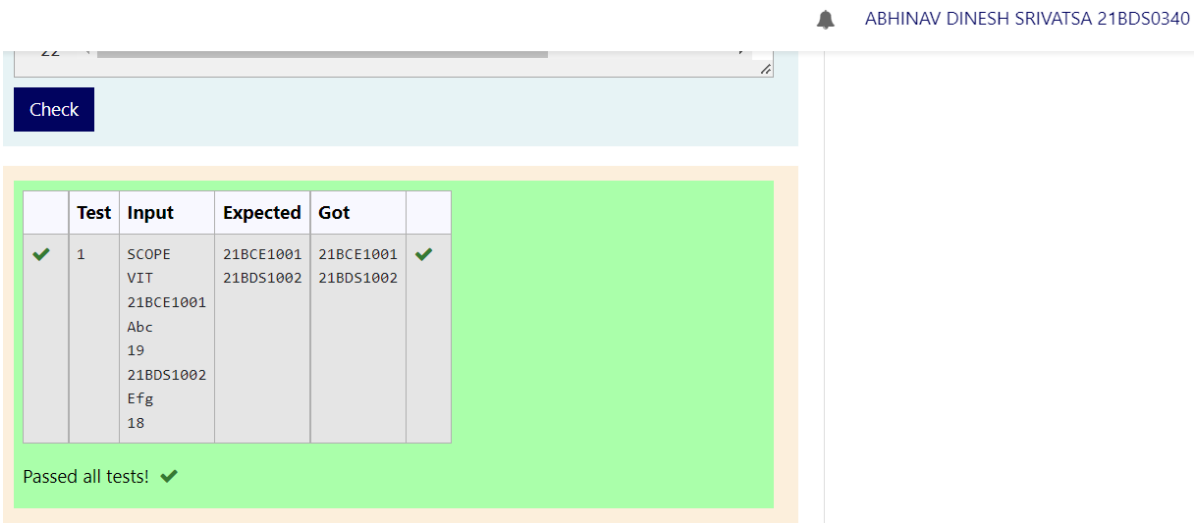
Call S[X]'s dispRegno()

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
#include <string.h>
using namespace std;

class Student
{
    char regno[10];
    char name[20];
    int age;

public:
    static char sec[10];
    static char uni[10];

    Student(char regno[10], char name[20], int age)
    {
        strcpy(this->regno, regno);
        strcpy(this->name, name);
        this->age = age;
    }
```

```
    void dispRegno()
    {
        cout << this->regno << "\n";
    }
};

char Student::sec[10];
char Student::uni[10];

int main()
{
    cin >> Student::sec;
    cin >> Student::uni;
    char regno[2][10];
    char name[2][20];
    int age[2];
    for (int x = 0; x < 2; x++)
        cin >> regno[x] >> name[x] >> age[x];
    Student s[2] = {Student(regno[0], name[0], age[0]), Student(regno[1],
name[1], age[1])};
    for (int x = 0; x < 2; x++)
        s[x].dispRegno();
}
```

Output



[Ex. No. M5_CSQ5]

AIM

Create a class student with necessary attributes specific to each student. Define few instances, invoke setValues and getValues member functions. Display each student's regno and phno through a member function, name displayInfo. Also create a non-member function to do the same task, name printInfo. [Function with object as an argument]

Algorithm / Pseudocode

Student:

Declare character array Regno with 10 spaces

Declare character array Name with 15 spaces

Declare integer Age

Void setValues(char R[10], char N[15], int A):

Assign Regno as R

Assign Name as N

Assign Age as A

Char* getRegno():

Return Regno

Char* getName():

Return Name

Int getAge():

Return Age

Void displayInfo():

Call getRegno(), getName(), getAge() and display

Void printInfo(Student S):

Call S's getRegno(), S's getName(), S's getAge() and display

Int main():

Declare Student S1 and S2

Declare character arrays Regno and Name with 10 and 15 spaces respectively

Declare integer Age

Read inputs and assign to Regno, Name and Age

Call S1's setValues() and pass Regno, Name and Age as arguments

Read inputs and assign to Regno, Name and Age

Call S2's setValues() and pass Regno, Name and Age as arguments

Call S1's displayInfo()

Call printInfo() and pass S2 as argument

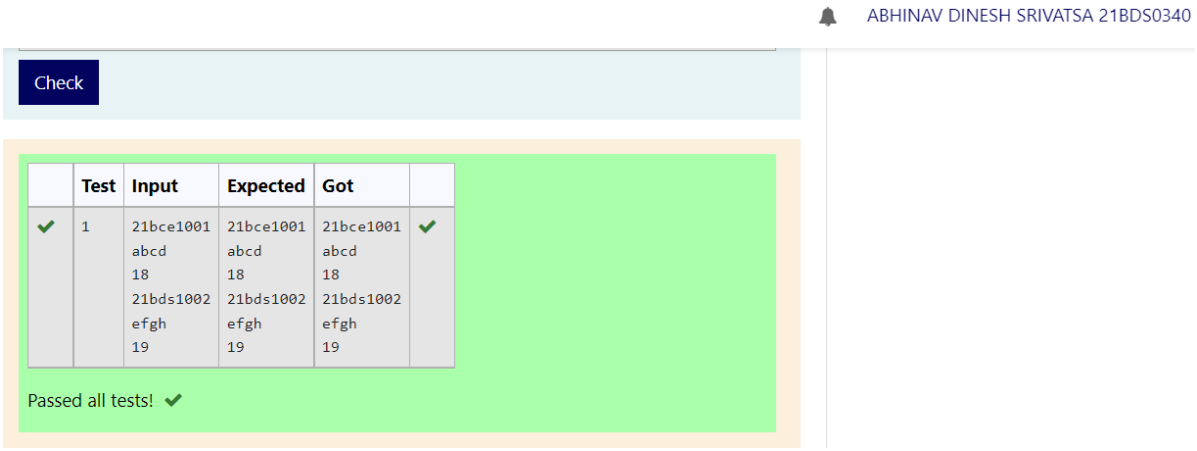
Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
#include <string.h>
using namespace std;

class Student
{
    char regno[10];
    char name[15];
    int age;

public:
    void setValues(char regno[10], char name[15], int age)
```

```
    {
        strcpy(this->regno, regno);
        strcpy(this->name, name);
        this->age = age;
    }

    char *getRegno()
    {
        return this->regno;
    }

    char *getName()
    {
        return this->name;
    }

    int getAge()
    {
        return this->age;
    }

    void displayInfo()
    {
        cout << this->getRegno() << "\n"
             << this->getName() << "\n"
             << this->getAge() << "\n";
    }
};

void printInfo(Student s)
{
    cout << s.getRegno() << "\n"
         << s.getName() << "\n"
         << s.getAge() << "\n";
}

int main()
{
    Student s1, s2;
    char regno[10], name[15];
    int age;
    cin >> regno >> name >> age;
    s1.setValues(regno, name, age);
    cin >> regno >> name >> age;
    s2.setValues(regno, name, age);
```

```
        s1.displayInfo();
        printInfo(s2);
    }
```

Output



[Ex. No. M5_CSQ6]

AIM

Define an inline function in CPP program for calculating factorial of n

Algorithm / Pseudocode

Int factorial(int N):

If N = 0, then return 1

Return N * factorial(N - 1)

Int main():

Declare integer N

Read input and assign to N

Call factorial() and pass N as argument and display

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
using namespace std;

inline int factorial(int n)
{
    if (n == 0)
        return 1;
    return n * factorial(n - 1);
}

int main()
{
    int n;
    cin >> n;
    cout << factorial(n);
}
```

Output


ABHINAV DINESH SRIVATSA 21BDS0340
21 }
22
Check
Test Input Expected Got
1 5 120 120
Passed all tests!

[Ex. No. M5_CSQ7]

AIM

Develop cpp function to swap two values

Algorithm / Pseudocode

Void swap(int *X, int *Y):

Declare integer pointer Temp and assign *X

Assign *X as *Y

Assign *Y as *Temp

Int main():

Declare integers X and Y

Read inputs and assign to X and Y

Call swap() and pass &X and &Y as arguments

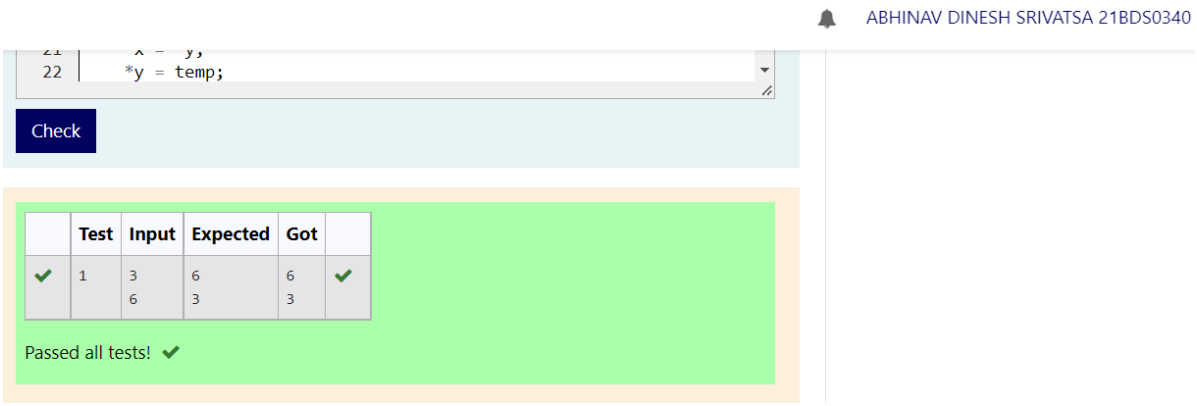
Display X and Y

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
using namespace std;

void swap(int *x, int *y)
{
    int temp = *x;
    *x = *y;
    *y = temp;
}

int main()
{
    int x, y;
    cin >> x >> y;
    swap(&x, &y);
    cout << x << "\n"
```

```
        << y << "\n";
    }
```

Output


ABHINAV DINESH SRIVATSA 21BDS0340
22 *y = temp;
Check
Test Input Expected Got
1 3 6 6 3 6 3
Passed all tests!

[Ex. No. M5_CSQ8]

AIM

Define cpp function to print -, 10 times if no argument passed, other print the character that is being passed along with times argument. If choice is 1, invoke print function with no argument, otherwise pass the arguments received as input.

Algorithm / Pseudocode

Void dispChar(int N = 10, Char C = '-'):

Loop from 0 to N as X

Display C

Int main():

Declare integer Choice

Read input and assign to Choice

If Choice = 1, then call dispChar()

Else

Declare character C

Declare integer N

Read inputs and assign to C and N

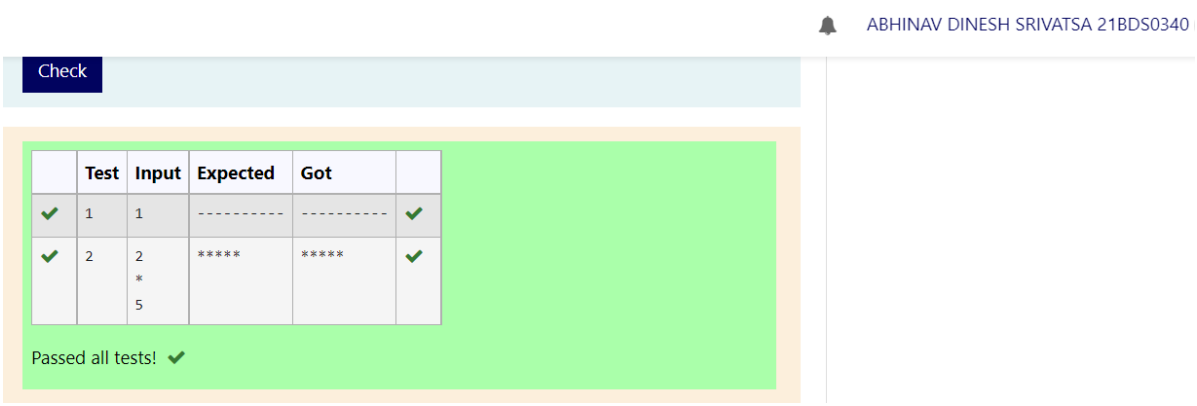
Call dispChar() and pass N and C as arguments

Program Code

```cpp
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
using namespace std;

void dispChar(int n = 10, char c = '-')
{
    for (int x = 0; x < n; x++)
        cout << c;
```

```
}

int main()
{
    int choice;
    cin >> choice;
    if (choice == 1)
        dispChar();
    else
    {
        char c;
        int n;
        cin >> c >> n;
        dispChar(n, c);
    }
}
```

Output



[Ex. No. M5_CSQ9]

AIM

Develop a friend function to calculate total fare for a ticket. There are 'n' passengers in the ticket and they are from a family of a railway employee. Discount is given for their travel based on the cader of the employee.

If Cader A - 10%, B - 15%, C - 20%

Algorithm / Pseudocode

Fare:

Declare integer Passengers

Declare character array From with 20 spaces

Declare character array To with 20 spaces

Declare integer Ticket_cost

Declare character Cader

Fare(int P, char F[20], char T[20], int Tc, char C):

Assign Passengers as P

Assign From as F

Assign To as T

Assign Ticket_cost as Tc

Assign Cader as C

Declare Price as friend class

Price:

Int calculate(Fare F):

Switch F's Cader

If 'A', then return F's Ticket_cost * F's Passengers * 0.9

If 'B', then return F's Ticket_cost * F's Passengers * 0.85

If 'C', then return F's Ticket_cost * F's Passengers * 0.8

Return 0

Int main():

Declare integers P and Tc

Declare character C and arrays Fr and T with 20 spaces

Read input and assign to P, Fr, T, Tc and C respectively

Declare Fare F() passing P, Fr, T, Tc and C as arguments

Declare Price Pr

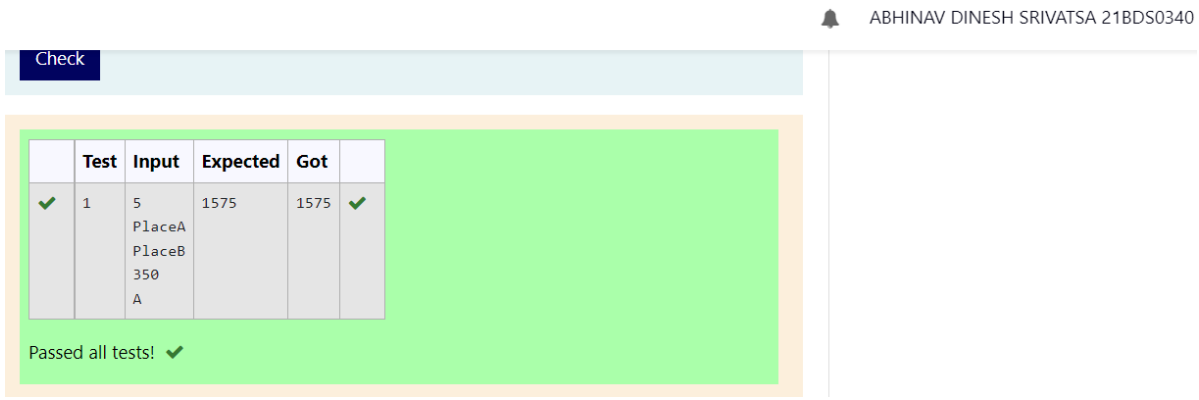
Call Pr's calculate() and pass F and display

Program Code

```
// 21BDS0340 Abhinav Dinesh Srivatsa

#include <iostream>
#include <string.h>
using namespace std;

class Fare
{
    int passengers;
    char from[20];
    char to[20];
    int ticket_cost;
    char cader;

public:
    Fare(int p, char f[20], char t[20], int tc, char c)
    {
        this->passengers = p;
        strcpy(this->from, f);
```

```
        strcpy(this->to, t);
        this->ticket_cost = tc;
        this->cader = c;
    }

    friend class Price;
};

class Price
{
public:
    int calculate(Fare f)
    {
        switch (f.cader)
        {
        case 'A':
            return f.ticket_cost * f.passengers * 0.9;
        case 'B':
            return f.ticket_cost * f.passengers * 0.85;
        case 'C':
            return f.ticket_cost * f.passengers * 0.8;
        }
        return 0;
    }
};

int main()
{
    int p, tc;
    char fr[20], t[20], c;
    cin >> p >> fr >> t >> tc >> c;
    Fare f(p, fr, t, tc, c);
    Price pr;
    cout << pr.calculate(f);
}
```

Output


ABHINAV DINESH SRIVATSA 21BDS0340
Check
Test
Input
Expected
Got
1
5
PlaceA
PlaceB
350
A
1575
1575
Passed all tests!

[Ex. No. M5_CSQ10]

AIM