CONVERT TEMPERATURE FROM DEGREE CELSIUS TO FAHRENHEIT

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

void  main()
{

   float celsius, fahrenheit;
   clrscr( );

   printf("\n \t  DEGREE CELSIUS TO FAHRENHEIT");
   printf(" \n \t  **********************************");

   printf(" Enter temperature in Celsius:  ");
   scanf("%f", &celsius);

   fahrenheit = ((1.8 * celsius) + 32);

   printf("\n\n Temperature in Fahrenheit : %f ", fahrenheit);

   getch( );

}
```

Output:

```
        DEGREE CELSIUS TO FAHRENHEIT
        **********************************
```

Enter temperature in Celsius: 32

Temperature in Fahrenheit : 89.59998

NUMBER IS ODD OR EVEN

```
#include <stdio.h>
#include<conio.h>
void  main()
{

   int  number;
   clrscr( );

   printf("\n \t EVEN OR ODD NUMBER");
   printf(" \n \t  **********************");
   printf(" Enter an integer:  ");
   scanf("%d", &number);

           if(number%2 ==0)
                printf("\n\n%d is Even Number. ", number);

           else
                 printf("\n\n %d is Odd Number. ", number);

  getch( );

}
```

Output:

```
        EVEN OR ODD NUMBER
        **********************

Enter an integer :   8

8 is Even Number.
```

GREATEST OF THREE NUMBERS

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

void  main()
{

   int  a, b, c;
   clrscr( );

  printf("\n \t GREATEST OF THREE NUMBERS");
  printf(" \n \t  ********************************");

  printf(" Enter a value of  a, b, c:  ");
   scanf(" %d %d %d ", &a, &b, &c);

            if(a > b && a > c)
            {
                 printf(" \n\n  A is greater than B and C. ");
             }
            else if(b > a && b > c)
            {
                 printf(" \n \n B is greater than A and C. ");
             }
            else if(c > a && c > b)
            {
                 printf(" \n\n C is greater than A and B. ");
             }
            else
            {
                 printf(" \n\n All are equal or any two values are equal. ");
             }

  getch( );

}
```

Output:

GREATEST OF THREE NUMBERS
****

Enter a value of a, b, c: 35 45 25

B is greater than A and C

DISPLAY FIRST TEN NATURAL NUMBERS AND THEIR SUM

```
#include <stdio.h>
#include<conio.h>
void  main()
{

   int  i, sum=0;
   clrscr( );

  printf("\n \t  FIRST TEN NATURAL NUMBERS");
  printf(" \n \t  ********************************");

 for(i=1; i<=10; i++)
   {
        printf("\n Number is %d. ", i);
        sum=sum+i;
    }
   printf("\n\n Sum = %d ", sum);
  getch( );

}
```

Output:

```
          FIRST TEN NATURAL NUMBERS
          ********************************
Number is  1
Number is  2
Number is  3
Number is  4
Number is  5
Number is  6
Number is  7
Number is  8
Number is  9
Number is  10

Sum = 55
```

DISPLAY DAYS FROM MONDAY TO SUNDAY

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

void  main()
{

   char  ch;
   clrscr( );

   printf("\n \t  DISPLAY DAYS FROM MONDAY TO SUNDAY");
   printf(" \n \t  ******************************************");

  printf("\n Enter M for Monday \n\t T for Tuesday \n\t W for Wednesday \n\t  H for
Thursday \n\t F for Friday \n\t S for Saturday \n\t U for Sunday: \n\n ");
 scanf("%c", &ch);

 switch(ch)
{

      case ‘m’:
      case ‘M’:
         printf("\t Monday");
         break;

      case ‘t’:
      case ‘T’:
         printf("\t Tuesday");
         break;

      case ‘w’:
      case ‘W’:
         printf("\t Wednesday");
         break;

      case ‘h’:
      case ‘H’:

         printf("\t Thursday");
         break;
```

```
        case ‘f’:
        case ‘F’:
            printf("\t Friday");
            break;

        case ‘s’:
        case ‘S’:
            printf("\t Saturday");
            break;

        case ‘u’:
        case ‘U’:
            printf("\t Sunday");
            break;


        default:
            printf("\t Invalid input");
            break;

    }

   getch( );
}
```

Output:

```
        DISPLAY DAYS FROM MONDAY TO SUNDAY
        ******************************************

Enter M for Monday
        T for Tuesday
        W for Wednesday
        H for Thursday
        F for Friday
       S for Saturday
       U for Sunday   :

    S       Sunday
```

MULTIPLICATION OF TWO MATRICES

```
#include <stdio.h>
#include <conio.h>
void  main( )
{

  int m, n, p, q, c, d, k, sum = 0;
  int first[10][10], second[10][10], multiply[10][10];
  clrscr( );

    printf("\n \t  MULTIPLICATION OF TWO MATRICES");
    printf(" \n \t  ************************************");

   printf("Enter number of rows and columns of first matrix:\n");
   scanf("%d %d", &m, &n);

   printf("Enter elements of first matrix:\n");

  for (c = 0; c < m; c++)
    for (d = 0; d < n; d++)
      scanf("%d", &first[c][d]);

  printf("Enter number of rows and columns of second matrix: \n");
  scanf("%d %d", &p, &q);

  if (n != p)
    printf("The matrices can't be multiplied with each other.\n");
  else
  {
    printf("Enter elements of second matrix:\n");

    for (c = 0; c < p; c++)
      for (d = 0; d < q; d++)
        scanf("%d", &second[c][d]);

    for (c = 0; c < m; c++)
{
```

```
      for (d = 0; d < q; d++)
{
       for (k = 0; k < p; k++)

 {
         sum = sum + first[c][k]*second[k][d];
}

    multiply[c][d] = sum;
    sum = 0;
 }
}

    printf("Product of the matrices: \n");

    for (c = 0; c < m; c++)
 {
      for (d = 0; d < q; d++)
        printf("%d\t", multiply[c][d]);

      printf("\n");
    }
  }

getch( );
}
```

Output:

```
        MULTIPLICATION OF TWO MATRICES
        **********************************

Enter the  number of rows and columns of first matrix:
3       3
Enter the element of  first matrix:
    1   2   0
    0   1   1
    2   0   1
Enter the  number of rows and columns of second matrix:
3       3
Enter the element of  second matrix:
    1    1   2
    2   1   1
    1   2   1
Product of  entered matrices:
  5   3   4
  3    3   2
  3    4   5
```

MAXIMUM NUMBER IN ARRAY USING POINTER

```
#include <stdio.h>
#include <conio.h>
void  main( )
{

   int a[10], n, i, max;
   int *p;
   clrscr( );

   printf("\n \t  MAXIMUM NUMBER IN ARRAY USING POINTER ");
   printf(" \n \t  ***********************************************");

  printf("Enter the size of aray: ");
  scanf("%d ", &n);

 printf("Enter %d elements in the array:\n", n);

 for (i = 0; i < n; i++)
        scanf("%d", &a[i]);
        printf("Elements in the array are: \n");

 for (i = 0; i < n; i++)
       printf("%5d", a[i]);
       p=&a[0];
       max=a[0];

 for (i = 0; i < n; i++)
      {
         if(max<=*p)
         max=*p;
         p++;
      }

printf("\n Maximum elements in the array is:   %d ", max);
getch( );
}
```

Output:

```
        MAXIMUM NUMBER IN ARRAY USING POINTER
        ********************************************

Enter the size of aray: 3
Enter %d elements in the array:
24
25
45
Elements in the array are:
24  25  45

Maximum elements in the array is: 45
```

REVERSE OF A NUMBER USING POINTER

```
#include<stdio.h>
#include<conio.h>
void  main()
{
   int n, a, *rev, *rem, *temp;
  clrscr(  );

   printf("\n \t  REVERSE OF A NUMBER USING POINTER");
   printf(" \n \t  ***************************************");
   printf("\n\n Enter any number: ");
   scanf("%d", &n);

  a=n;
  temp= &n;
  *rev=0;

While(*temp > 0)
{
*rem = *temp%10;
*temp = *temp/10;
*rev = (*rev)*10+ *rem;
}

   printf("\n Reverse of %d is  = %d" , a, *rev);
   getch( );
}
```

Output:

```
        REVERSE OF A NUMBER USING POINTER
        ***************************************

 Enter any  number: 459
 Reverse of 459 is  = 954
```

FACTORIAL OF NUMBER USING RECURSION

```
#include<stdio.h>
#include<conio.h>
int factorial(int);
void  main()
{
   int num;
  clrscr(  );

   printf("\n \t  FACTORIAL OF NUMBERS USING RECURSION");
   printf(" \n \t  ****************************************");
   printf("\n\n Enter any integer number: ");
   scanf("%d", &num);

   printf("\n Factorial of %d is: %d", num, factorial(num));
   getch( );
}

int factorial(int i)
{
    int f;

   if(i==1)
      return 1;
   else
      f=i*factorial(i-1);

      return f;
}
```

Output:

```
        FACTORIAL OF NUMBERS USING RECURSION
        ****************************************
Enter any integer number: 4
Factorial of 4 is: 24
```

ADD TWO NUMBERS USING POINTER

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

void  main()
{

    int  first, second, *p, *q, sum;
    clrscr( );

   printf(("\n \t ADD TWO NUMBERS USING POINTER");
   printf(" \n \t  ************************************");

   printf("\n\n  Enter two integers to add :  ");
   scanf(" %d  %d", &first, &second);

   p=&first;
  q=&second;
  sum=*p+*q;

printf(" \n\n  Sum of numbers = %d " sum);
getch( );

}
```

Output:

```
        ADD TWO NUMBERS USING POINTER
        ************************************

 Enter two integers to add :  50  25

Sum of numbers = 75
```

CALL BY VALUE AND CALL BY REFERENCE

```
#include<stdio.h>
#include<conio.h>
void  call_by_value(int a)
{
      a+=10;
      printf("\n Inside Call_by_value a = %d", a);
}

void  call_by_reference(int *b)
{
      (*b)+=10;
      printf("\n Inside Call_by_reference b = %d", *b);
}

void  main()
{
   int a=10, b=10;
   clrscr(  );

   printf(("\n \t  CALL BY VALUE AND CALL BY REFERENCE");
   printf(" \n \t  ****************************************");

   printf("\n\n CALL BY VALUE ");
   printf("\n ~~~~~~~~~~~~~~ ");
   printf("\n Before Function Call a = %d", a);
  call_by_value(a);
  printf("\nAfter Function Call a = %d", a);

   printf("\n\n CALL BY REFERENCE");
   printf("\n ~~~~~~~~~~~~~~~~~~ ");
   printf("\n Before Function Call b = %d", b);
  call_by_value(&b);
  printf("\n After Function Call b = %d", b);

 getch( );
}
```

Output:

```
        CALL BY VALUE AND CALL BY REFERENCE
        ***********************************************

 CALL BY VALUE
 ~~~~~~~~~~~~~~~~
 Before Function Call a = 10
 Inside Call_by_value a =20
 After Function Call a = 10

 CALL BY REFERENCE
 ~~~~~~~~~~~~~~~~~~~~
 Before Function Call b = 10
Inside Call_by_reference b = 20
After Function Call b = 20
```

CREATE A FILE CONTAINING STUDENT DETAILS

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

struct s
{
        int regno, age;
        char name[50], dob[20], sex[10], course[25];
};

void  main( )
{
        Struct s a[5], b[5];
        FILE *fptr;
        int i, n;
        clrscr( );
        fptr=fopen(" file.txt ", " wb ");
       printf("\n CREATE A FILE CONTAINING STUDENT DETAILS ");
       printf("\n ********************************************");
       printf ("\n Enter the number of students:  ");
       scanf(" %d ", &n);

for(i=0; i<n ; ++i)
    {
          fflush(stdin);
          printf(" \n Enter Name : ");
          gets(a[i].name);
          printf(" \n Enter Course: ");
          gets(a[i].course);
          printf(" \n Enter DOB : ");
          gets(a[i].dob);
          printf(" \n Enter Sex: ");
          gets(a[i]. sex);
          printf(" \n Enter Register Number : ");
          scanf (" %d ", &a[i].regno);
          printf(" \n Enter Age: ");
          scanf (" %d ", &a[i]. age);
}
```

```
fwrite(a, sizeof(a), 1, fptr);
fclose(fptr);
printf(" \n %d Files created Successfully ", n);

fptr=fopen(" file.txt ", " rb "  );
fread(b, sizeof(b), 1, fptr);

printf(" \n\n ---------------------------------------------------- ");
printf(" \n\n Reg No \t  Name \t Age \t DOB \t Sex \t Course ");
printf(" \n\n ---------------------------------------------------- ");

for(i=0; i<n; ++i)
{
            printf(" \n\n %d \t  %s \t %d \t %s \t %s \t %s ", b[i].regno, b[i].name, b[i].age,
          b[i].dob, b[i].course);
}

printf(" \n\n ---------------------------------------------------- ");

fclose(fptr);

getch( );
}
```

Output:

CREATE A FILE CONTAINING STUDENT DETAILS

****

Enter the number of students: 2

Enter Name : Dharshini B

Enter Course: CS

Enter DOB : 21/07/2001

Enter Sex: Female

Enter Register Number : 101

Enter Age: 17

Enter Name : Rithish V

Enter Course: CS

Enter DOB : 02/02/2001

Enter Sex: Male

Enter Register Number : 102

Enter Age: 17

Reg No	Name	Age	DOB	Sex	Course
101	Dharshini B	17	21/07/2001	Female	CS
102	Rithish V	17	02/02/2001	Male	CS

UPDATE A FILE CONTAINING STUDENT DETAILS

```
#include <stdio.h>
#include<conio.h>
#include <stdlib.h>
#include<string.h>
#define size  200

struct stu
{
        int  id;
        char *name;
}  *stu1, *stu2;

void display( );
void create( );
void update( );
FILE *fp, *fp1;
int count = 0;

void  main( char  **argv)
{

        int i, n, ch;
        clrscr( );

       printf("\n UPDATE A FILE CONTAINING STUDENT DETAILS ");
       printf("\n ********************************************");

       printf ("\n\n1. Create a Record");
       printf ("\n2. Display a Record");
       printf ("\n3. Update a Record");
       printf ("\n4. Exit");

       while(1)
            {
                printf ("\n\n Enter your choice:  ");
                scanf(" %d ", &ch);
```

```
switch(ch)
{
case 1:
            fp=fopen(argv[1], " a ");
            create( );
            break;
case 2:
            fp1=fopen(argv[1], " r ");
            display( );
            break;
case 3:
            fp1=fopen(argv[1], " r ");
            update( );
            break;
case 4:
            exit( 0 );
            break;

            }
      }
}

void create( )
{
          int  i;
          char *p;
          stu1= (struct stu *) malloc(sizeof(struct stu));
          stu1->name= (char *) malloc((size) * (sizeof(char)));

          printf(" Enter the  name of student:  ");
          scanf(" %[^\n]s " , stu1-> name);
          printf(" Enter the student regno:  ");
          scanf(" %d " , &stu1-> id);

          fwrite(&stu1-> id, sizeof(stu1-> id, 1, fp);
          fwrite(stu1 -> name, size, 1, fp);
          count++;
          fclose(fp);
}
```

```
void display( )
{
        int   i=1;
        stu2 = (strust stu *) malloc (1 * sizeof(struct stu));
        stu2->name = (char *) malloc(size*sizeof(char));

        if(fp1 == NULL)
                printf(" \n File not opened for reading ");

        while( i <= count)
        {
                fread(&stu2->id, sizeof(stu2->id), 1, fp1);
                fread(stu2->name, size, 1, fp1);
                printf("\n %d %s", stu2 -> id, stu2-> name);
                i++;
        }

        fclose(fp1);
        free(stu2->name);
        free(stu2);
        }

        void update( )
        {
                int   id, flag = 0, i=1;
                char s[ size];

        if(fp1 == NULL)

        {
                 printf("\n File cannot be opened.);
                return;
        }
                printf("\n Enter Student Reg.No to update");
                scanf(" %d " , & id);

                stu2= (struct stu *) malloc (1 * sizeof(struct stu));
                stu2-> name= (char *) malloc(size * sizeof(char));
```

```
while(i<=count)
{
    fread(&stu2 ->id , sizeof (stu2->id), 1, fp1);
    fread(stu2 -> name, size, 1, fp);

if(id == stu2->id)
    {
        printf("\n Enter new name of students to update: " );
        scanf(" %[^\n]s " , s);

        fseek(fp1, -204L, SEEK _ CUR);
        fwrite(&stu2->id, sizeof(stu2->id), 1, fp1);
        fwrite(s, size, 1,fp1);
        flag =1;
        break;
   }

i++;
}

if(flag ! =1)
    {
        printf(" No student record found.");
        flag = 0;
    }

fclose(fp1);
free(stu 2-> name);
free(stu2);
}
```

Output:

UPDATE A FILE CONTAINING STUDENT DETAILS
****

1. Create a Record
2. Display a Record
3. Update a Record
4. Exit

Enter your choice : 1

Enter the name of student: Charu

Enter the student regno: 200

Enter your choice : 1

Enter the name of student: Deepika

Enter the student regno: 201

Enter your choice : 1

Enter the name of student: Harsha

Enter the student regno: 202

Enter your choice : 1

Enter the name of student: Rohith

Enter the student regno: 203

Enter your choice : 1

Enter the name of student: Varsha

Enter the student regno: 204

Enter your choice : 2

200 Charu

201 Deepika

202 Harsha

203 Rohith

205 Varsha

Enter your choice : 3

Enter student regno to update : 201

Enter new name of student to update : Dharani

Enter your choice : 2

200 Charu

201 Dharani

202 Harsha

203 Rohith

205 Varsha

Enter your choice: 4

QUADRATIC EQUATION

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

void main()
{
float a,b,c,root1,root2,rpl,num,imag;
double k,d;
clrscr();
printf("\n\n\t\t\tQUADRATIC EQUATION");
printf("\n\t\t\t********************");
printf("\nEnter the value of A: ");
scanf("%f",&a);
printf("\nEnter the value of B: ");
scanf("%f",&b);
printf("\nEnter the value of C: ");
scanf("%f",&c);
d=((b*b)-(4*a*c));
if(d==0)
{
printf("\n\n\nTHE ROOTS ARE REAL AND EQUAL\n");
root1=(-b/(2*a));
root2=root1;
printf("\n\nTHE VALUES OF ROOT1=%f AND ROOT2=%f",root1,root2);
}
```

```
else
if(d>0)
{
k=sqrt(d);
printf("\n\n\nTHE ROOTS ARE REAL AND UNEQUAL\n");
root1=((-b+k)/(2*a));
root2=((-b-k)/(2*a));
printf("\nTHE VALUES OF ROOT1=%f AND ROOT2=%f",root1,root2);
}
if(d<0)
{
printf("\n\n\nTHE ROOTS ARE REAL AND IMAGINARY");
rpl=(-b/(2*a));
d=-d;
num=pow((double)d,(double)0.5);
imag=num/(2*a);
printf("\n\nROOT1=%f+i%f",rpl,imag);
printf("\n\nROOT2=%f-i%f",rpl,imag);
}
getch();
}
```

OUTPUT:

QUADRATIC EQUATION

****

Enter the value of A: 2

Enter the value of B: 4

Enter the value of C: 2

THE ROOTS ARE REAL AND EQUAL

THE VALUES OF ROOT1= -1.000000 AND ROOT2= -1.000000

Enter the value of A: 2

Enter the value of B: 4

Enter the value of C: 1

THE ROOTS ARE REAL AND UNEQUAL

THE VALUES OF ROOT1=-0.292893 AND ROOT2 = -1.707107

DEPARTMENT OF COMPUTER SCIENCE

ANNAI WOMEN'S COLLEGE (ARTS & SCIENCE)

(Affiliated to Bharathidasan University – Trichirappalli-21)

Punnamchatram , Karur- 639 136.

Bonafide Certificate

Name :

Class : I – B.Sc (CS) **Reg.No :**

Subject : Programming in C Lab **Subject Code : 16SCCCS1P**

Certified that the record of the work done in the computer lab during the odd semester and submitted for the practical examination held on _____.

Staff Incharge **Head of the Department**

Examiners

1.

2.

ANNAI WOMEN'S COLLEGE (ARTS & SCIENCE)

(**Affiliated to Bharathidasan University, Tiruchirappalli)**

PUNNAMCHATRAM , KARUR.

PROGRAMMING IN C

B.SC (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE

I – SEMESTER

NOVEMBER-2021

CONTENTS