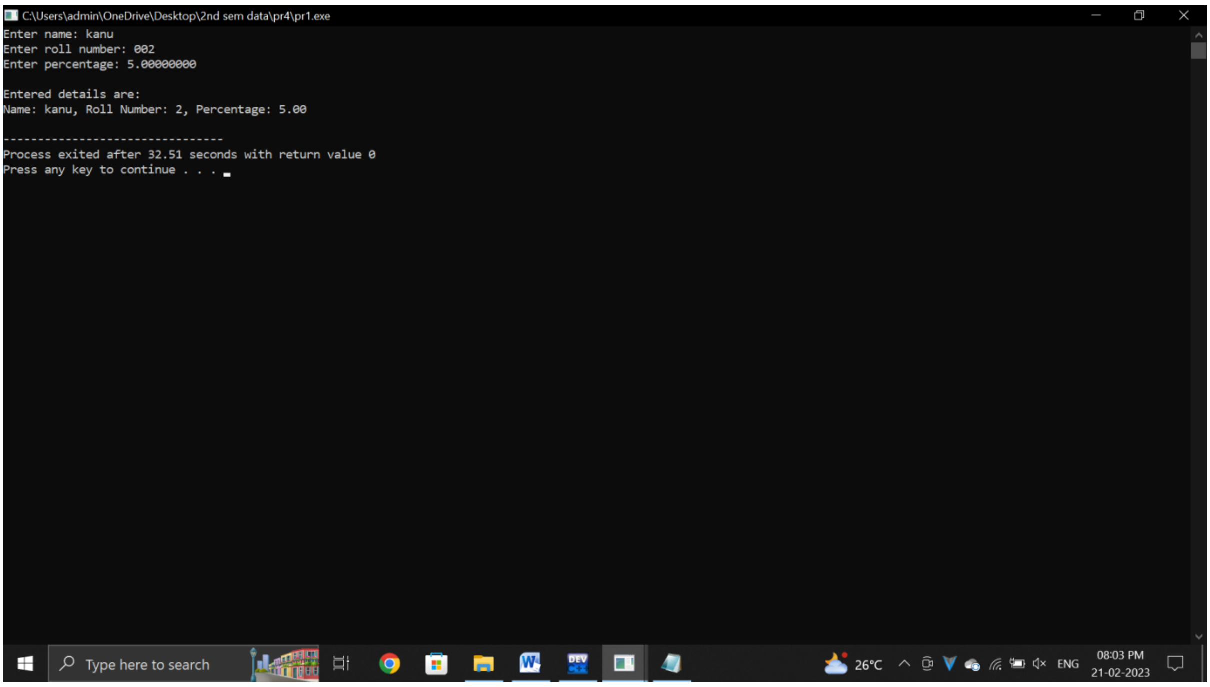
Practical 1:- Write a c program to read and print the student details using structure and Dynamic memory Allocation. Using malloc().

Input:-

```
#include <stdio.h>

#include <stdlib.h>


struct student

{

    char name[30];

    int roll;

    float perc;

};


int main()

{

    struct student *pstd;


    pstd=(struct student*)malloc(1*sizeof(struct student));


    if(pstd==NULL)

    {
        printf("Insufficient Memory, Exiting... \n");
        return 0;
    }

    printf("Enter name: ");
    gets(pstd->name);
    printf("Enter roll number: ");
    scanf("%d",&pstd->roll);
    printf("Enter percentage: ");
    scanf("%f",&pstd->perc);

    printf("\nEntered details are:\n");
    printf("Name: %s, Roll Number: %d, Percentage: %.2f\n",pstd->name,pstd->roll,pstd->perc);

    return 0;
}
```

Output:-



C:\Users\admin\OneDrive\Desktop\2nd sem data\pr4\pr1.exe

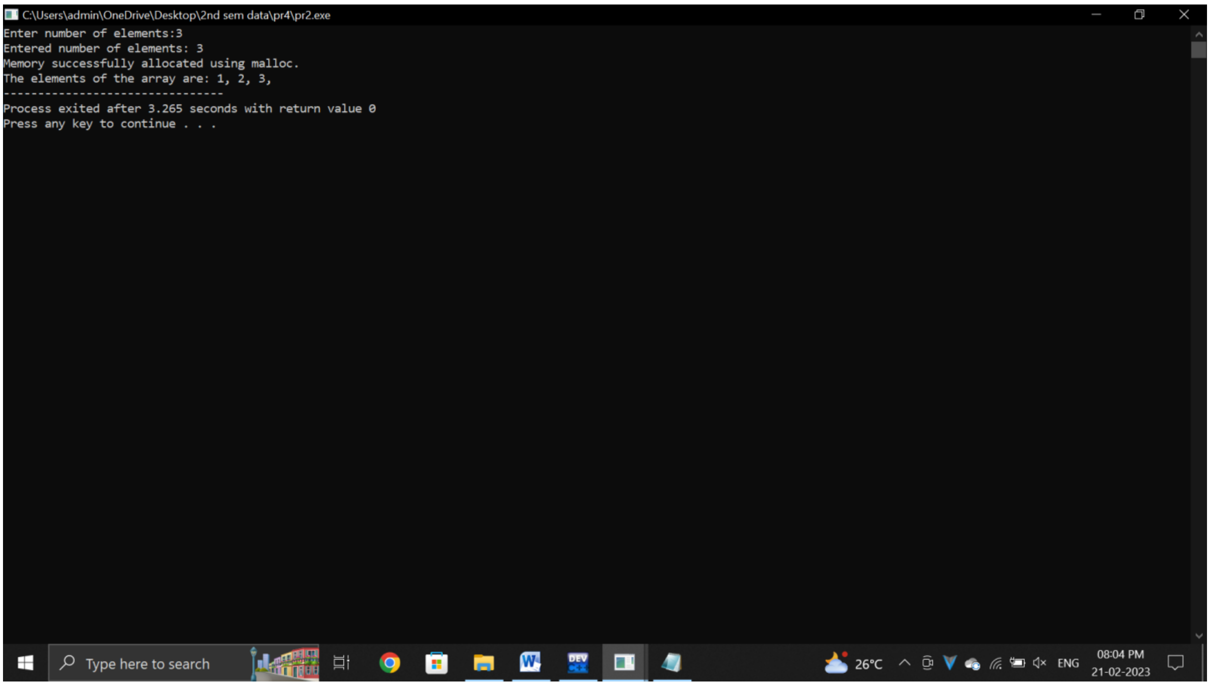
```
Enter name: kanu
Enter roll number: 002
Enter percentage: 5.00000000

Entered details are:
Name: kanu, Roll Number: 2, Percentage: 5.00

--------------------------------
Process exited after 32.51 seconds with return value 0
Press any key to continue . . .
```

Practical 2:- Write a program in c to create a float array dynamically using calloc().input values and display.

Input:-

```
#include <stdio.h>

#include <stdlib.h>


int main()

{

    int* ptr;

    int n, i;


    printf("Enter number of elements:");

    scanf("%d",&n);

    printf("Entered number of elements: %d\n", n);
    ptr = (int*)malloc(n * sizeof(int));

    if (ptr == NULL) {
        printf("Memory not allocated.\n");
        exit(0);
    }
    else {

        printf("Memory successfully allocated using malloc.\n");

        for (i = 0; i < n; ++i) {
            ptr[i] = i + 1;
        }

        printf("The elements of the array are: ");
        for (i = 0; i < n; ++i) {
            printf("%d, ", ptr[i]);
        }
    }

    return 0;
}
```

Output:-



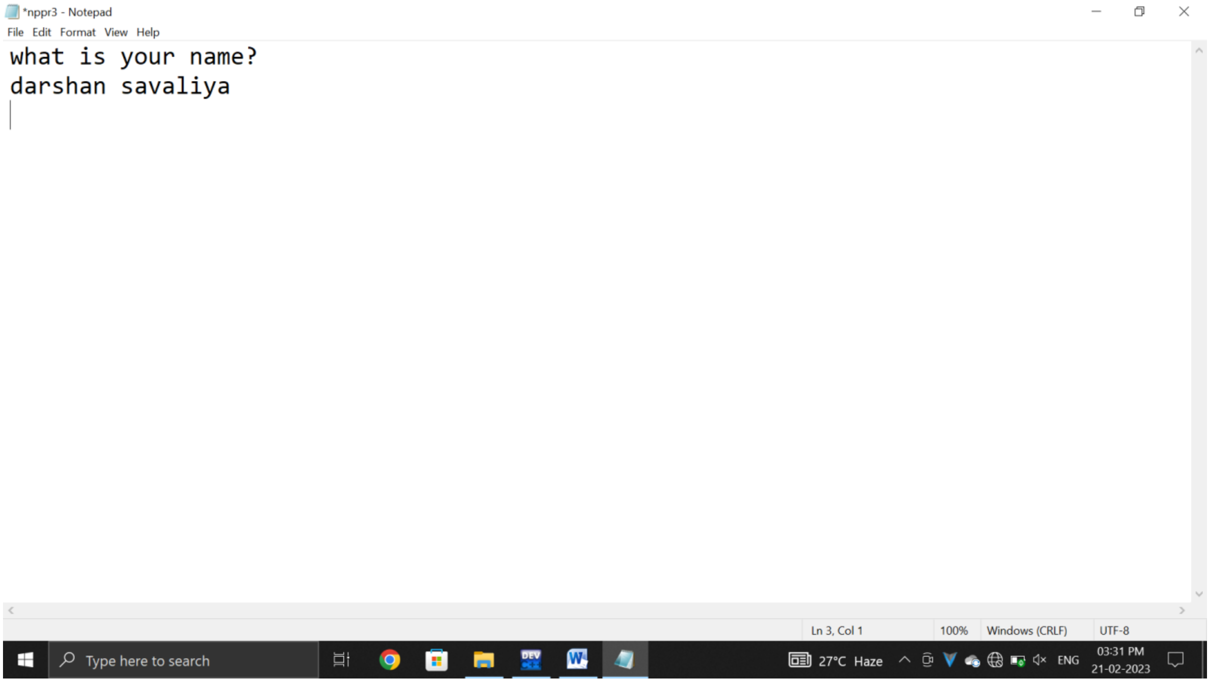
```
Enter number of elements:3
Entered number of elements: 3
Memory successfully allocated using malloc.
The elements of the array are: 1, 2, 3,
--------------------------------
Process exited after 3.265 seconds with return value 0
Press any key to continue . . .
```

Practical 3:- Write a Program to open a file in append mode and add new records in it.

Input:-

```
#include<stdio.h>

#include<stdlib.h>

int main(){

        FILE *fp;

        fp=fopen("nppr3.txt","w");

        fputs("what is your name? \n",fp);

        fclose(fp);

}
```

Output:-



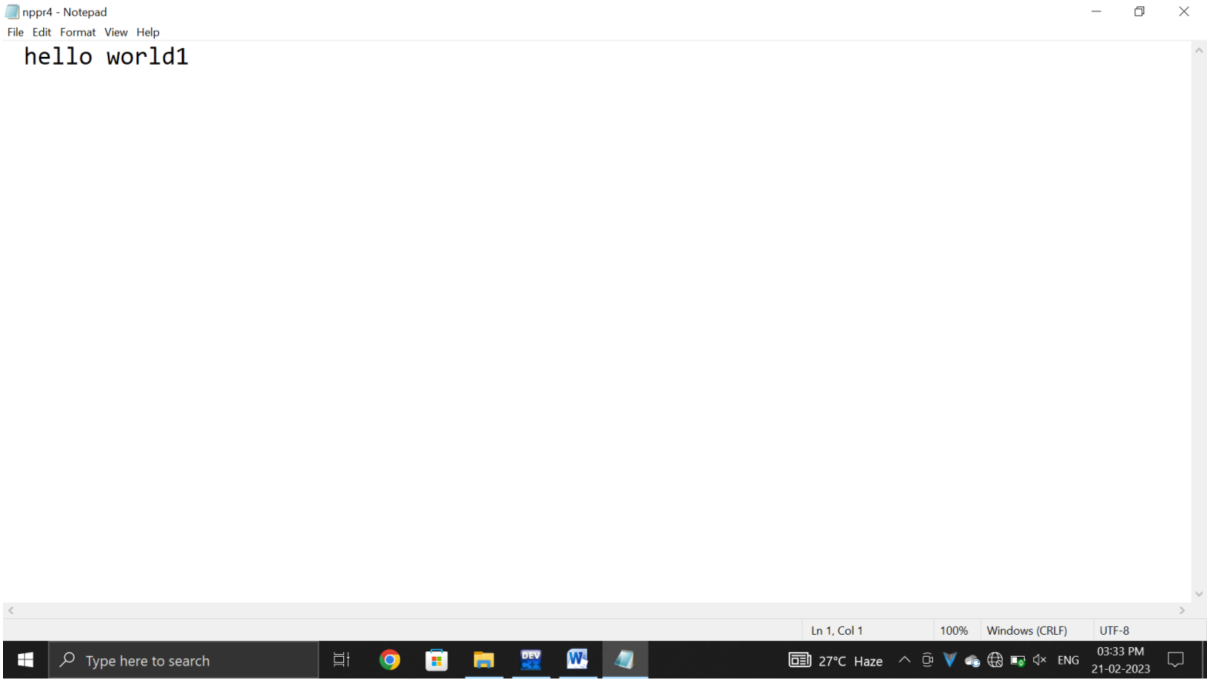
```
what is your name?
darshan savaliya
```

Practical 4:- Write a Program to open a file in read/write mode in it. Read and Write new information in the file

Input:-

```
#include<stdio.h>

#include<stdlib.h>

int main(){


        FILE *fp;

        char c;

        fp=fopen("nppr4.txt","w+");
        fputs(" hello world1 \n",fp);
        fp=fopen("nppr4.txt","r+");

        while(1){
                c=fgetc(fp);
                if(c==EOF);
                break;
                printf("%c",c);

        }
        fclose(fp);

}
```

Output:-



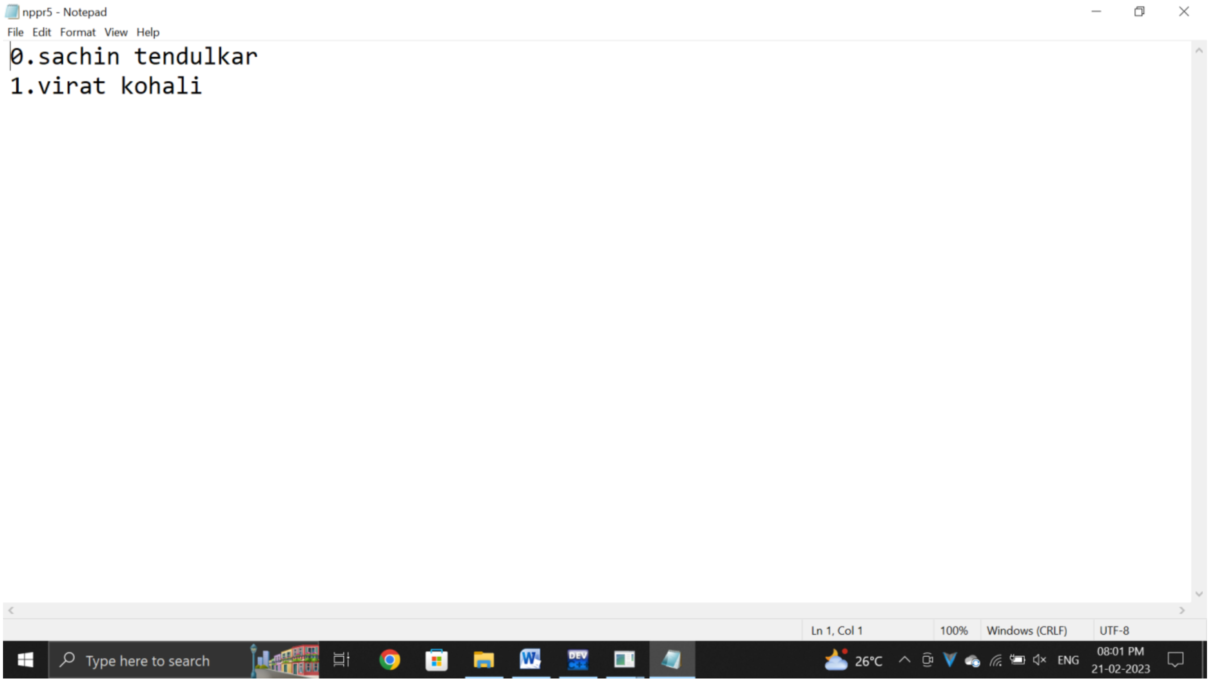
```
 hello world1
```

Practical 5:- Write a Program to open a file and write some text using fprintf() function. Open the file and verify the contents.

Input:-

```
#include<stdio.h>

int main()

{

        int i, n=2;

        char str[50];


        FILE *fptr = fopen("nppr5.txt", "w");

        if (fptr == NULL)

        {

                printf("Could not open file");

                return 0;

        }


        for (i = 0; i < n; i++)

        {

                puts("Enter a name");

                scanf("%[^\n]%*c", str);

                fprintf(fptr,"%d.%s\n", i, str);

        }

        fclose(fptr);


        return 0;
}
```

Output:-



```
0.sachin tendulkar
1.virat kohali
```

Practical 6:- A file name data contains series of integer numbers. Write a c program to read all numbers from file and then write all odd numbers into file named "odd" and write all even numbers into file named "even".

Input:-

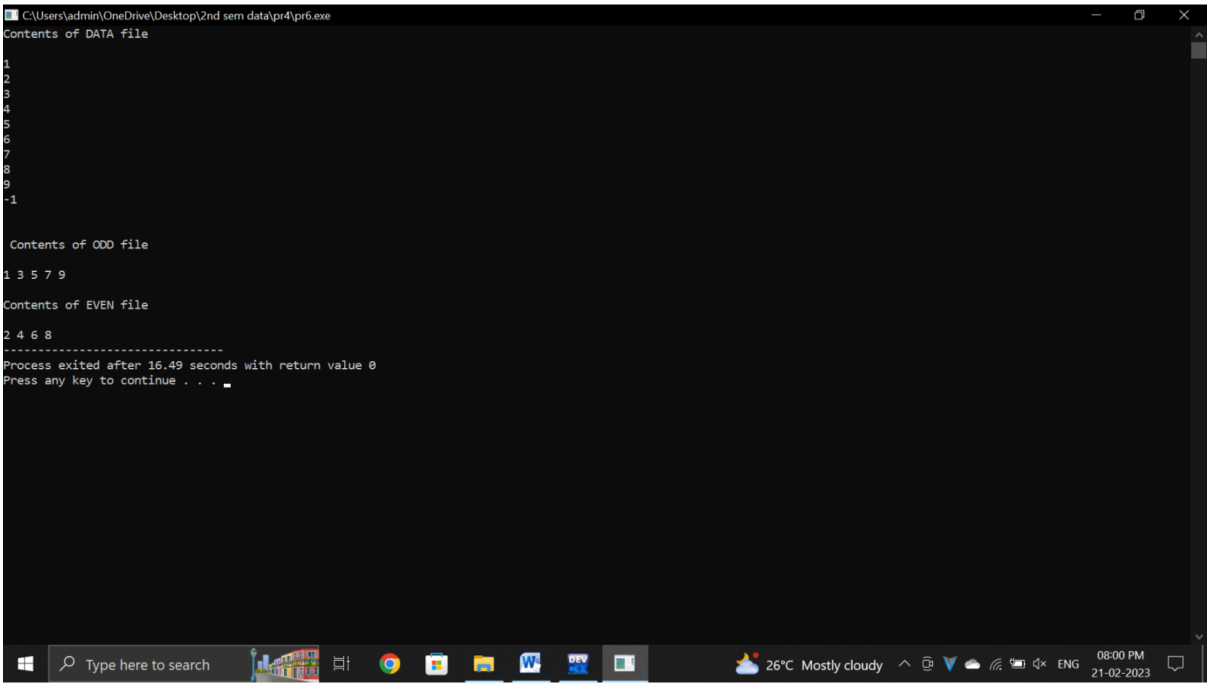
```
#include<stdio.h>


int main()

{

   FILE *f1,*f2,*f3;

   int number,i, n=10;
```

```
printf("Contents of DATA file\n\n");


f1 = fopen("DATA","w");


for(i=0;i<n;i++)
{
   scanf("%d",&number);
   if(number==-1)
   {
        break;
   }
   putw(number,f1);
}
fclose(f1);


f1 = fopen("DATA","r");
f2 = fopen("ODD","w");
f3 = fopen("EVEN","w");


while((number = getw(f1)) != EOF)
{
   if(number%2==0)
{
putw(number,f3);
}
   else
```

```
 {
  putw(number,f2);
 }
  }


  fclose(f1);
  fclose(f2);
  fclose(f3);


  f2 = fopen("ODD","r");
  f3 = fopen("EVEN","r");


  printf("\n\n Contents of ODD file \n\n");


  while((number = getw(f2)) != EOF)
{
 printf("%d ",number);
}



  printf("\n\nContents of EVEN file \n\n");


  while((number = getw(f3)) != EOF)
{
 printf("%d ",number);
}
```

```
    fclose(f2);

    fclose(f3);

    return 0;

}
```

Output:-



```
Contents of DATA file

1
2
3
4
5
6
7
8
9
-1


 Contents of ODD file

1 3 5 7 9

Contents of EVEN file

2 4 6 8
--------------------------------
Process exited after 16.49 seconds with return value 0
Press any key to continue . . .
```