PSPC ASSIGNMENT

NAME : RITAV KASHYAP

ASTU ROLL NO : 200610003078

SEMESTER : 2ND SEC : D

COLLEGE ROLL NO : 20/237

BRANCH : ELECTRICAL ENGINEERING

Problem No. 1:

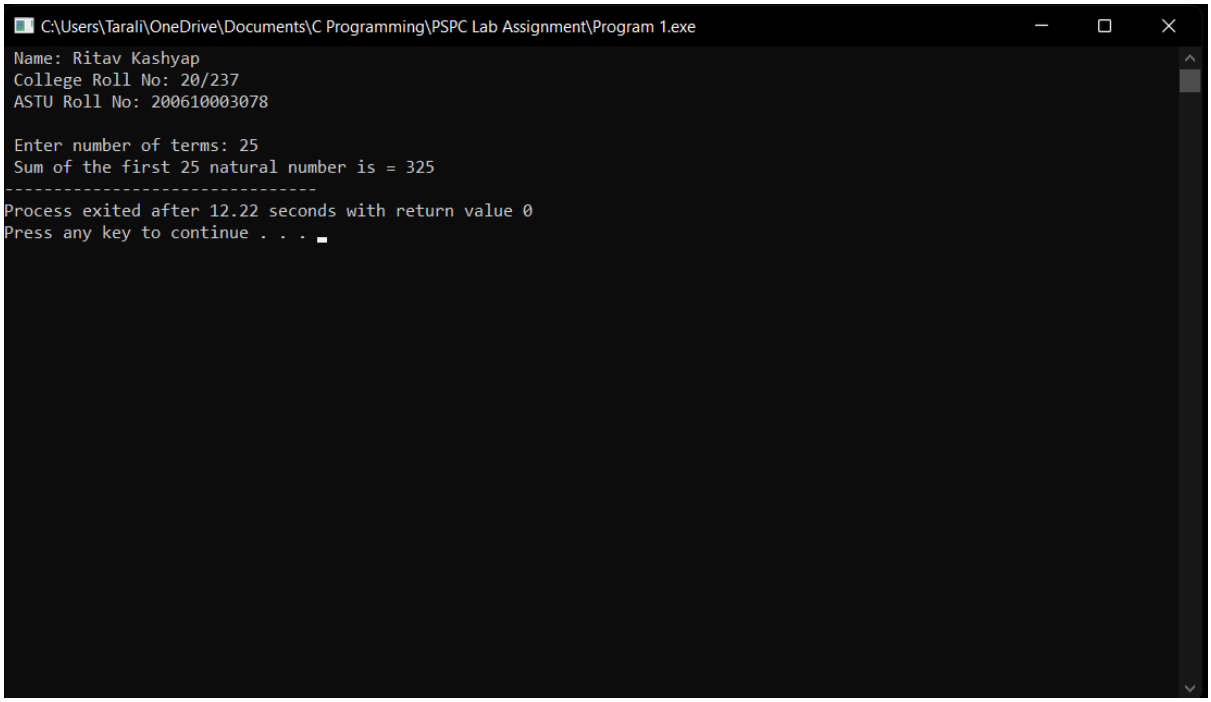
Aim: Create a mathematical series and write a recursive function to calculate the sum of the series.

Program:

```
/* Program No: 1
   Aim: Create a mathematical series and write a recursive function to calculate
        the sum of the series.
   Name: Ritav Kashyap
   ASTU Roll No: 200610003078 */

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

int addNumbers (int n);

int main ()
{
        printf (" Name: Ritav Kashyap\n");
        printf (" College Roll No: 20/237\n");
        printf (" ASTU Roll No: 200610003078\n\n");

        int num;
        printf (" Enter number of terms: ");
        scanf ("%d", &num);
        printf (" Sum of the first %d natural number is = %d", num,
```

```
                addNumbers(num));
        return 0;
}

int addNumbers (int n)
{
      if (n != 0)
           return n + addNumbers (n - 1);
      else
          return n;
}
```

Sample output:



Problem No. 2:

Aim: Use Dynamic Memory Allocation to create a 1D array and take input of numbers and use a function to perform selection sort.

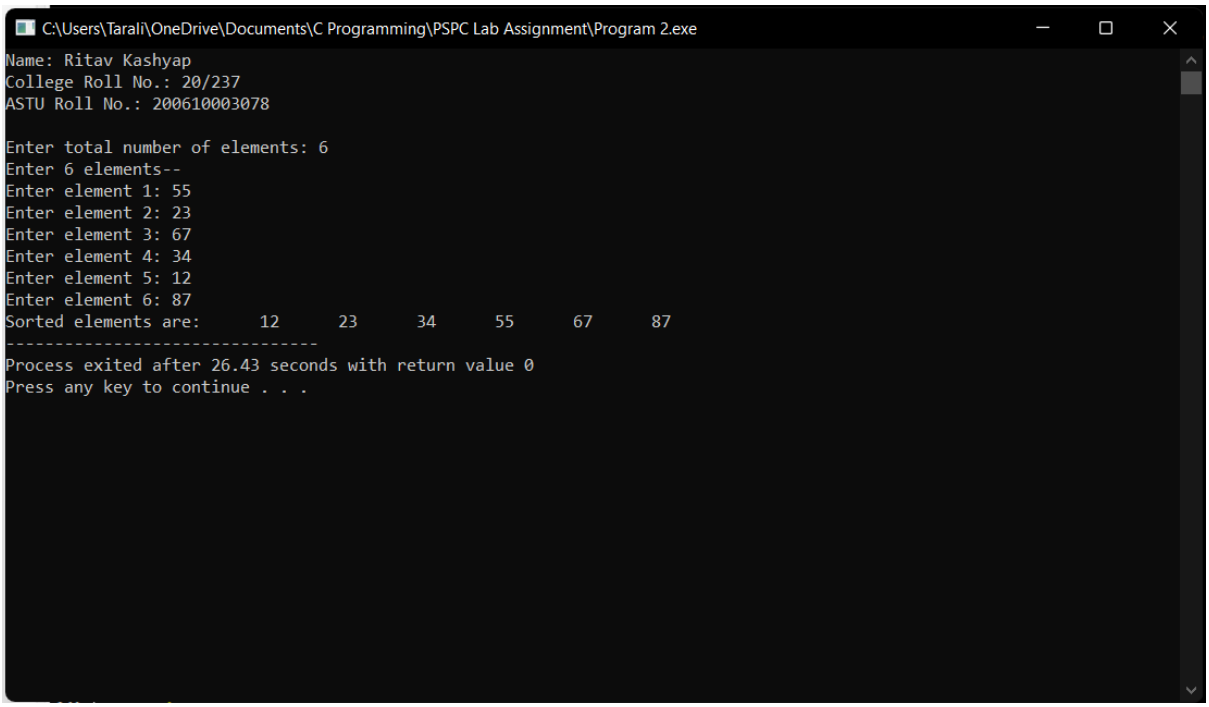
Program:

```
/* Program No: 2
   Aim: Use Dynamic Memory Allocation to create a 1D array and take input of
        numbers and use a function to perform selection sort.
   Name: Ritav Kashyap
   ASTU Roll No: 200610003078 */

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

void swap (int *xp, int *yp)
{
    int temp = *xp;
    *xp = *yp;
    *yp = temp;
}

void selectionSort (int *arr, int n)
{
    int i, j, min_idx;
    for (i=0; i<n-1; i++)
          {
        min_idx= i;
```

```
        for (j= i+1; j<n; j++)
        {
          if (arr[j] < arr[min_idx])
          {
                    min_idx = j;
                              }
                    }
        swap(&arr[min_idx], &arr[i]);
    }
}

int main ()
{
    int *arr;
    int lim, i;

    printf (" Name: Ritav Kashyap\n");
    printf (" College Roll No: 20/237\n");
    printf (" ASTU Roll No: 200610003078\n\n");

    printf ("Enter total number of elements: ");
    scanf ("%d", &lim);
    arr=(int*) malloc(lim*sizeof(int));

    printf ("Enter %d elements--\n", lim);
    for (i=0; i<lim; i++)
         {
```

```
        printf ("Enter element %d: ", i+1);
        scanf ("%d”, (arr+i));
    }

    selectionSort (arr, lim);
    printf ("Sorted elements are:");
    for (i=0; i<lim; i++)
    {
         printf ("%8d”, *(arr+i));
         }
    free(arr);
    return 0;
}
```

Sample output:



Problem No. 3:

Aim: Use pointer notation to create two 2D matrix, take input of data and perform matrix multiplication.

Program:

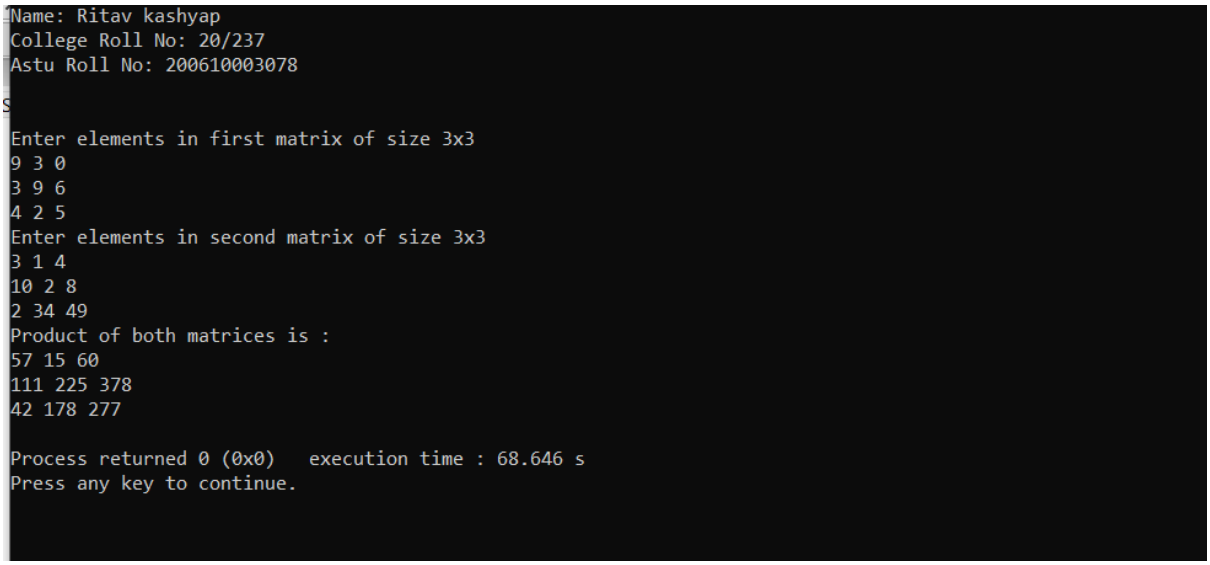
```
/* Program No: 3
   Aim: Use pointer notation to create two 2D matrix, take input of data and
        perform matrix multiplication.
   Name: Ritav Kashyap
   ASTU Roll No: 200610003078*/

#include <stdio.h>
#include<stdlib.h>
#define ROW 3
#define COL 3

void matrixInput (int mat [] [COL]);
void matrixPrint (int mat [] [COL]);
void matrixMultiply (int mat1[] [COL], int mat2[] [COL], int res [] [COL]);

int main ()
{

    printf (" Name: Ritav Kashyap\n");
    printf (" College Roll No: 20/237\n");
    printf (" Astu Roll No: 200610003078\n\n");
```

```
    int mat1[ROW][COL];
    int mat2[ROW][COL];
    int product [ROW][COL];

    printf ("Enter elements in first matrix of size %dx%d\n", ROW, COL);
    matrixInput(mat1);

    printf ("Enter elements in second matrix of size %dx%d\n", ROW, COL);
    matrixInput(mat2);

    matrixMultiply (mat1, mat2, product);
    printf ("Product of both matrices is: \n");
    matrixPrint(product);

    return 0;
}

void matrixInput (int mat [] [COL])
{
    int row, col;

    for (row = 0; row < ROW; row++)
    {
        for (col = 0; col < COL; col++)
        {
            scanf ("%d", (*(mat + row) + col));
        }
```

```
    }
}

void matrixPrint (int mat [] [COL])
{
    int row, col;
    for (row = 0; row < ROW; row++)
    {
        for (col = 0; col < COL; col++)
        {
            printf ("%d ", *(*(mat + row) + col));
        }
        printf("\n");
    }
}

void matrixMultiply (int mat1[] [COL], int mat2[] [COL], int res[] [COL])
{
    int row, col, i;
    int sum;


    for (row = 0; row < ROW; row++)
    {
        for (col = 0; col < COL; col++)
        {
            sum = 0;
```

```
            for (i = 0; i < COL; i++)
            {
                sum += (*(*(mat1 + row) + i)) * (*(*(mat2 + i) + col));
            }
            *(*(res + row) + col) = sum;
        }
    }
}
```

Sample output:



Problem No. 4:

Aim: Use Dynamic Memory Allocation to create a 1D array and take input of numbers and use a function to perform binary search.

Program:

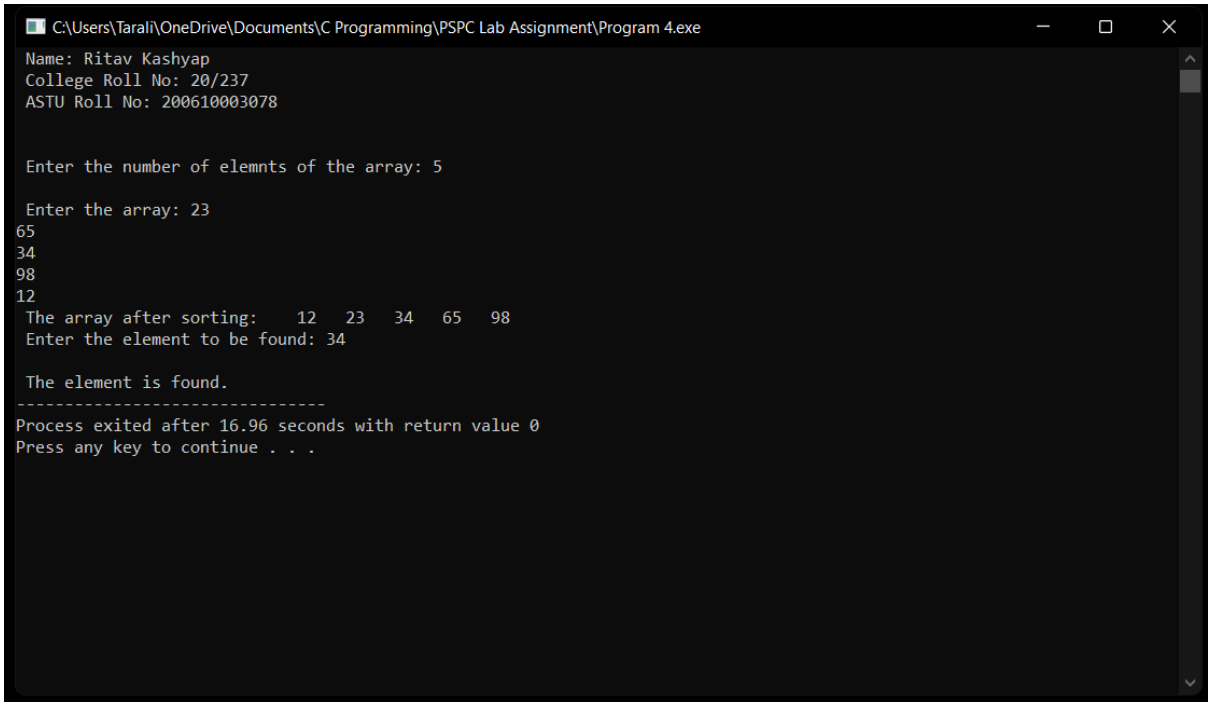
```
/* Program No: 4
   Aim: Use Dynamic Memory Allocation to create a 1D array and take input of
        numbers and use a function to perform binary search
   Name: Ritav Kashyap
   ASTU Roll No: 200610003078 */

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

int main ()
{
    int *p, n, i, k;
    int binarysr (int *p, int n);

    printf (" Name: Ritav Kashyap\n");
    printf (" College Roll No: 20/237\n");
    printf (" ASTU Roll No: 200610003008\n\n");

    printf ("\n Enter the number of elemnts of the array: ");
    scanf ("%d", &n); p=(int*) malloc(n*sizeof(int));
    printf ("\n Enter the array: ");
```

```
for (i=0; i<n; i++)
{
   scanf ("%d", (p+i));
}

int min, j, pos, temp;
for (i=0; i<n-1; i++)
     {
   min=i;
   for (j=i+1; j<n; j++)
   {
      if(p[j]<p[min])
      {
         min=j;
      }
   }
   temp=p[i];
   p[i]=p[min];
   p[min]=temp;
}

printf (" The array after sorting: ");
for (i=0; i<n; i++)
{
   printf ("%5d", p[i]);
}
```

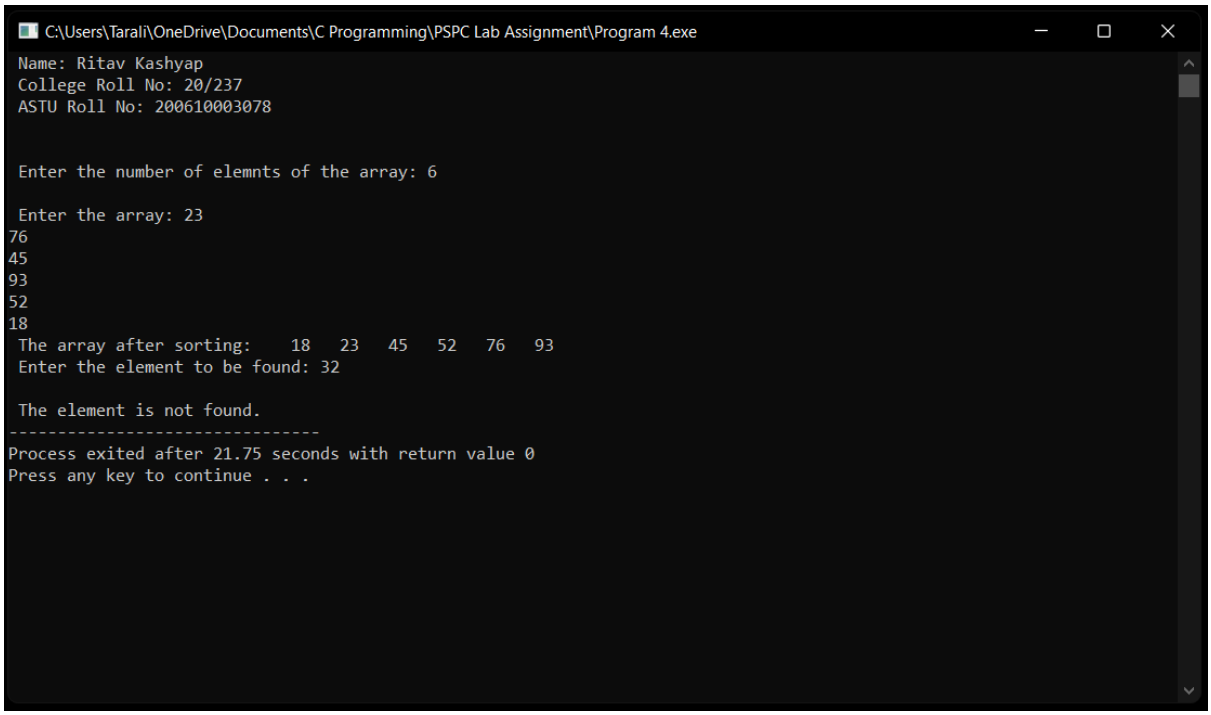
```
    binarysr (p, n);
    return 0;
}
int binarysr (int *p, int n)
{
    int k, beg=0, mid, end;
    printf ("\n Enter the element to be found: ");
    scanf ("%d", &k);
    end=n-1;
    while(beg<=end)
    {
        mid= (end + beg)/2;
        if(p[mid]==k)
        {
            printf ("\n The element is found.");
            exit (0);
        }
        else if (p[mid]>k)
        {
            end=mid-1;
        }
        else beg=mid+1;
    }
    printf ("\n The element is not found.");
}
```

Sample output:

If the element is found--



If the element is not found--



Problem No. 5:

Aim: Use File Handling mechanism and switch statement to write, append and display information about an employee in a company.

Program:

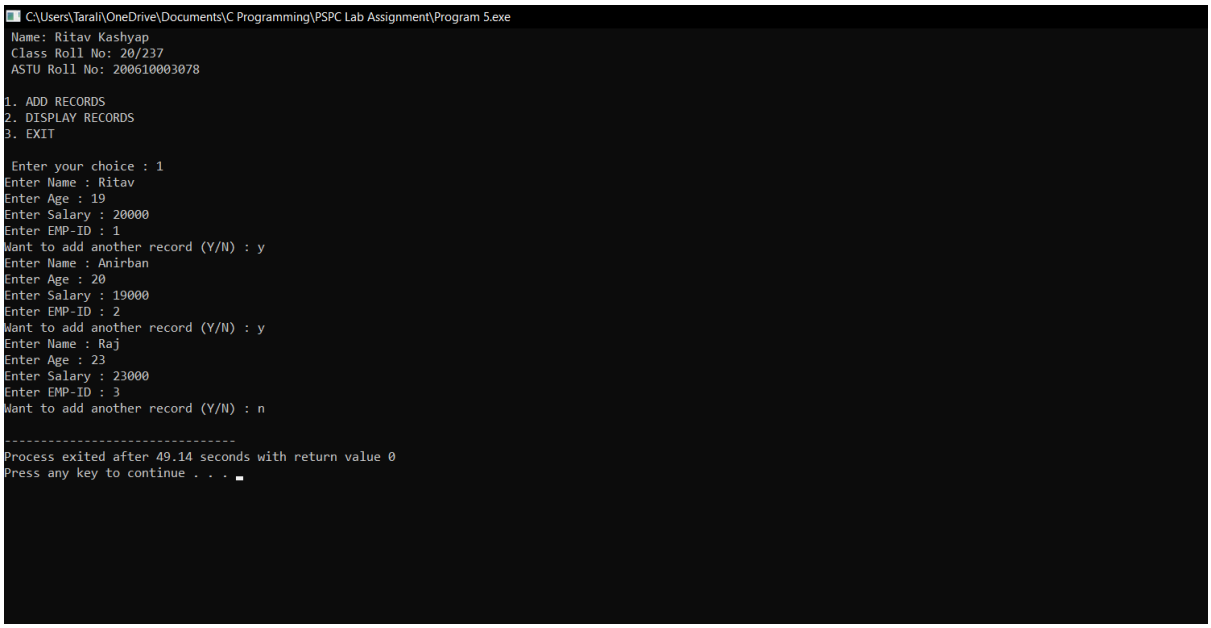
```
/* Program No: 5
   Aim: Use File Handling mechanism and switch statement to write, append
        and display information about an employee in a company.
   Name: Ritav Kashyap
   ASTU Roll No: 200610003078*/

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

struct employee
{
    char name [50];
    float salary;
    int age;
    int id;
};

struct employee E;
long int size = sizeof(E);
FILE *fp;

void addrecord ()
```

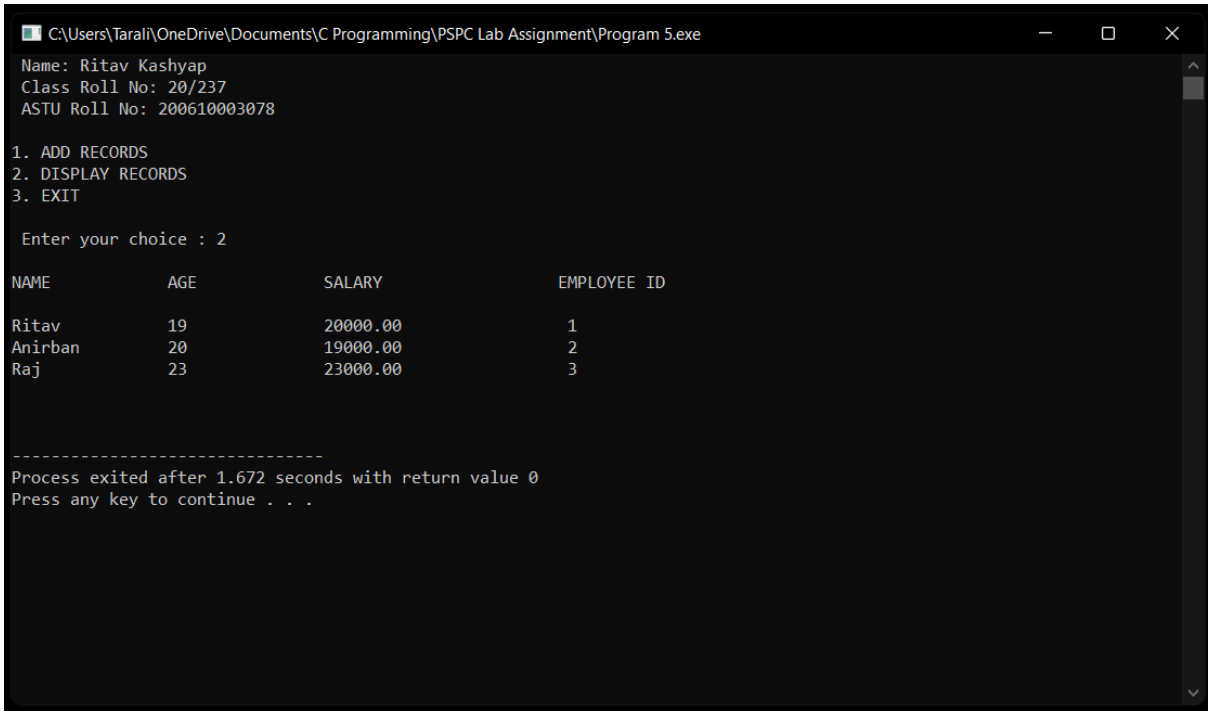
```
{
   char ch = 'y';
   while (ch == 'y')
   {
      printf ("Enter Name: ");
      scanf ("%s", E.name);
      printf ("Enter Age: ");
      scanf ("%d", &E.age);
      printf ("Enter Salary: ");
      scanf ("%f", &E.salary);
      printf ("Enter EMP-ID: ");
      scanf ("%d", &E.id);
      fwrite (&E, size, 1, fp);
      printf ("Want to add another record (Y/N): ");
      fflush(stdin);
      scanf ("%c", &ch);
   }
}

void displayrecord ()
{
   printf ("\nNAME\t\tAGE\t\tSALARY\t\t\tEMPLOYEE ID\n", E.name, E.age,
            E.salary, E.id);
   while (fread (&E, size, 1, fp) == 1)
   printf ("\n%s\t\t%d\t\t%.2f\t%10d", E.name, E.age, E.salary, E.id);
   printf("\n\n\n\t");
}
```

```
int main ()
{
   int opt;
   printf (" Name: Ritav Kashyap\n");
   printf (" Class Roll No: 20/237\n");
   printf (" ASTU Roll No: 200610003078\n\n");

   fp = fopen ("data.txt", "a+b");
   printf ("1. ADD RECORDS\n");
   printf ("2. DISPLAY RECORDS\n");
   printf ("3. EXIT\n\n");
   printf (" Enter your choice: ");
   scanf ("%d", &opt);

   switch(opt)
   {
      case 1:
      addrecord ();
      break;

      case 2:
      displayrecord ();
      break;

      case 3:
      fclose(fp);
```

```
        exit (0);
        break;

        default:
        printf ("invalid number");
        break;
    }
   return 0;
}
```

Sample output:




C:\Users\Tarali\OneDrive\Documents\C Programming\PSPC Lab Assignment\Program 5.exe
Name: Ritav Kashyap
Class Roll No: 20/237
ASTU Roll No: 200610003078
1. ADD RECORDS
2. DISPLAY RECORDS
3. EXIT
Enter your choice : 2
NAME AGE SALARY EMPLOYEE ID
Ritav 19 20000.00 1
Anirban 20 19000.00 2
Raj 23 23000.00 3
Process exited after 1.672 seconds with return value 0
Press any key to continue . . .