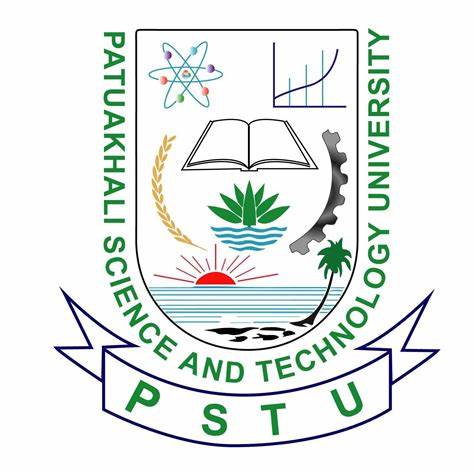
**PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY**



# **Course Code: CIT-111/112**

**SUBMITTED TO:**  
 **Prof.****Md. Mahbubur Rahman**

### **Department of Computer Science and Information Technology Faculty of Computer Science and Engineering**

**SUBMITTED BY:**

Name: **Nazmus Sakib**

ID: **2102066**, Registration No: **10193**

**Faculty of Computer Science and Engineering**

Date of submission: **18th August,2023.**

# Assignment**: Basic Code-09: File, Structure and Union + W3 resources- Searching and Sorting, File Handling.**

1. Write a program to read data from the keyboard, write it to a file called **INPUT,** again read the same data from the **INPUT** file, and display it on the screen.

#include<stdio.h>

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*f1;

char c;

printf("Data Input\n\n");

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

while((c=getchar()) != EOF)

putc(c,f1);

fclose(f1);

printf("\nData Output\n\n");

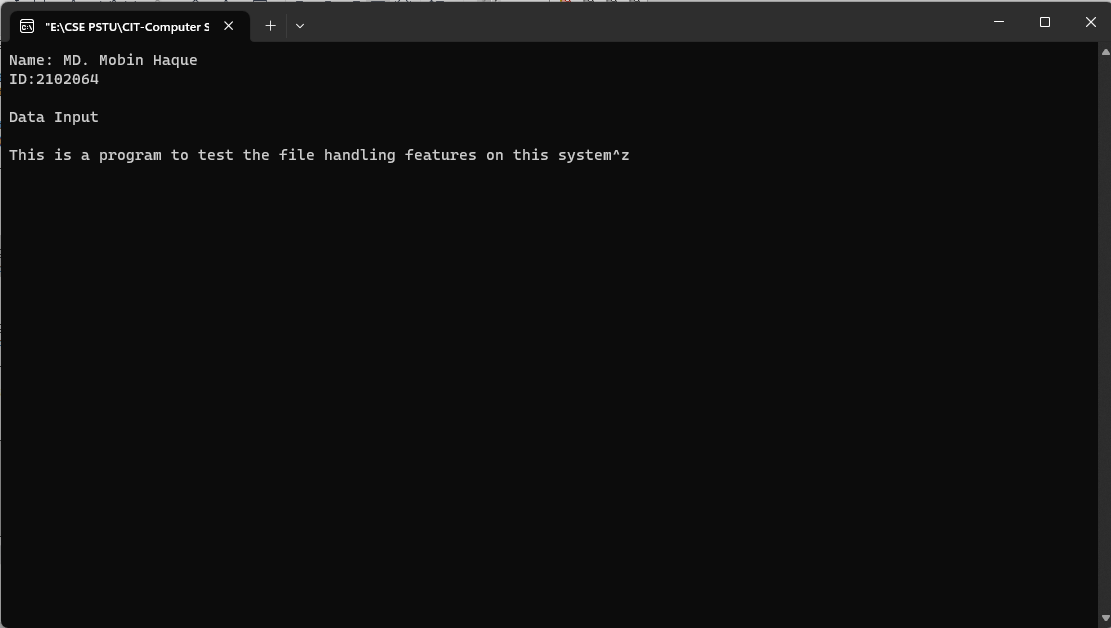
f1 = fopen("INPUT","r");

while((c=getc(f1)) != EOF)

printf("%c",c);

fclose(f1);

}



1. A file named **DATA** contains a series of integer numbers. Code a program to read these numbers and then write all 'odd' numbers to a file to be called **ODD** and all `even' numbers to a file to be called **EVEN.**

#include<stdio.h>

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*f1, \*f2, \*f3;

int number, i;

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

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

for(i = 1; i <= 30; 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\nContents of ODD file\n\n");

while((number = getw(f2)) != EOF)

printf("%4d", number);

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

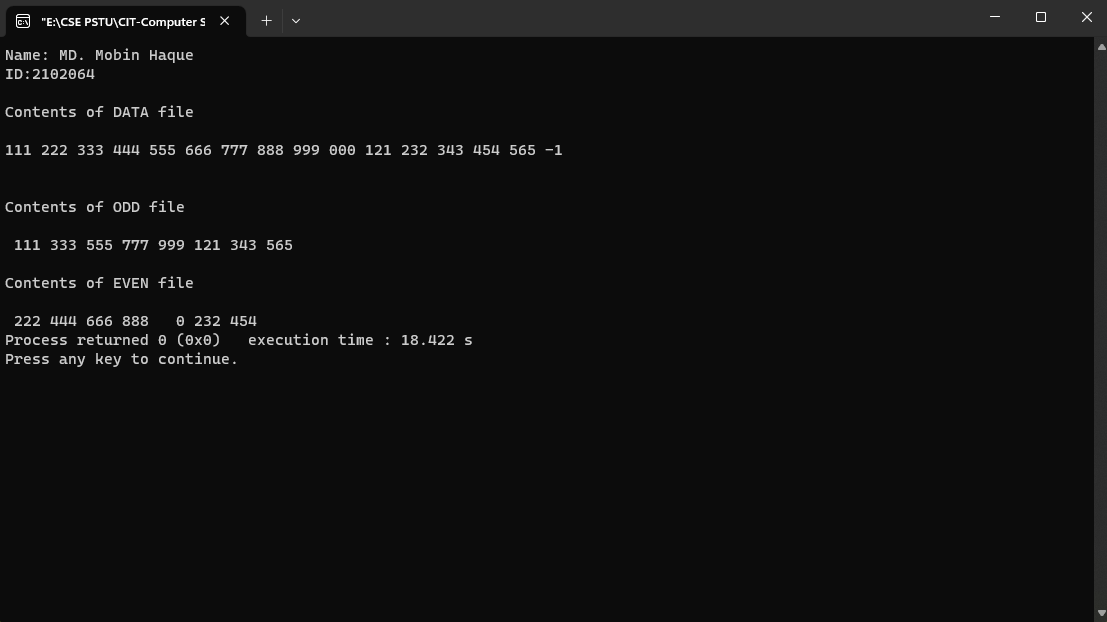
while((number = getw(f3)) != EOF)

printf("%4d", number);

fclose(f2);

fclose(f3);

}



1. Write a program to open a file named INVENTORY and store in it the following data:

**Item name Number Price Quantity**

AAA-1 111 17.50 115

BBB-2 125 36.00 75

C-3 247 31.75 104

Extend the program to read this data from the file INVENTORY and display the inventory table with the value of each item.

#include<stdio.h>

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*fp;

int number, quantity, i;

float price, value;

char item[10], filename[10];

printf("Input file name\n");

scanf("%s", filename);

fp = fopen(filename, "w");

printf("Input inventory data\n\n");

printf("Item name Number Price Quantity\n");

for(i = 1; i <= 3; i++)

{

fscanf(stdin, "%s %d %f %d",

item, &number, &price, &quantity);

fprintf(fp, "%s %d %.2f %d",

item, number, price, quantity);

}

fclose(fp);

fprintf(stdout, "\n\n");

fp = fopen(filename, "r");

printf("Item name Number Price Quantity Value\n");

for(i = 1; i <= 3; i++)

{

fscanf(fp, "%s %d %f d",item,&number,&price,&quantity);

value = price \* quantity;

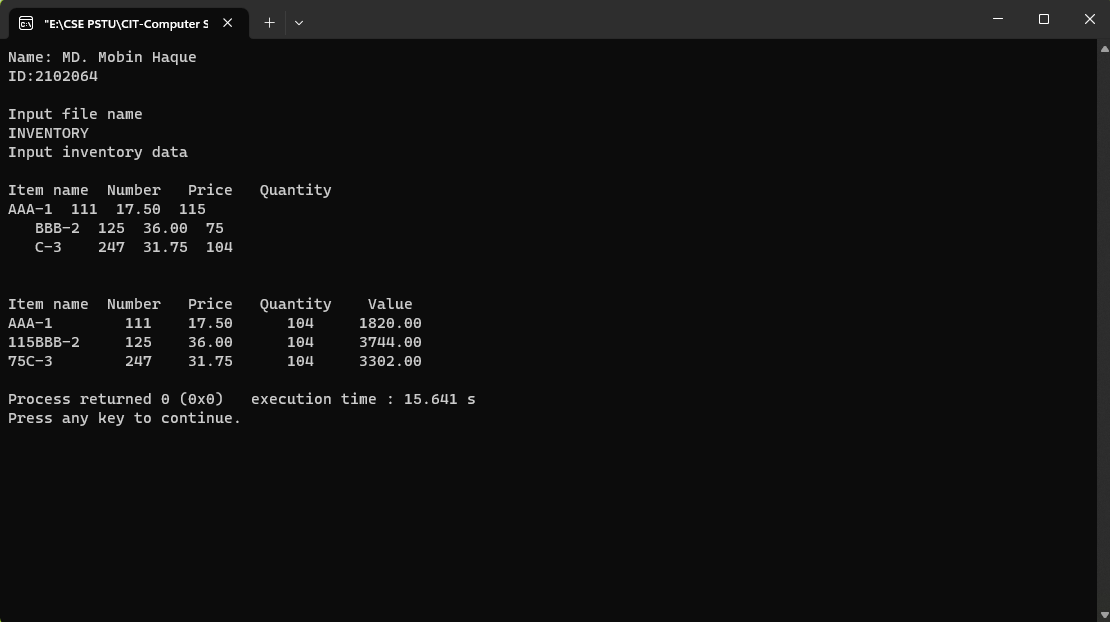
fprintf(stdout, "%-8s %7d %8.2f %8d %11.2f\n",

item, number, price, quantity, value);

}

fclose(fp);

}

****

1. Write a program to illustrate error handling in file operations.

#include<stdio.h>

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

char \*filename;

FILE \*fp1, \*fp2;

int i, number;

fp1 = fopen("TEST", "w");

for(i = 10; i <= 100; i += 10)

putw(i, fp1);

fclose(fp1);

printf("\nInput filename\n");

open\_file:

scanf("%s", filename);

if((fp2 = fopen(filename,"r")) == NULL)

{

printf("Cannot open the file.\n");

printf("Type filename again.\n\n");

goto open\_file;

}

else

for(i = 1; i <= 20; i++)

{ number = getw(fp2);

if(feof(fp2))

{

printf("\nRan out of data.\n");

break;

}

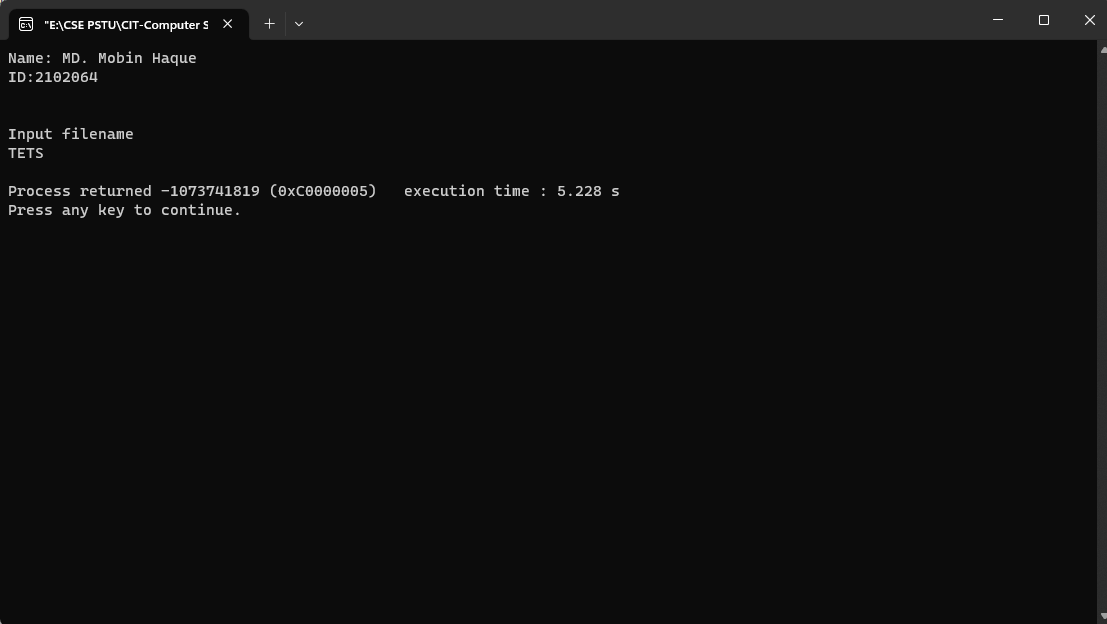
else

printf("%d\n", number);

}

fclose(fp2);

}

****

1. Write a program that uses the functions **ftell** and **fseek.**

#include<stdio.h>

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*fp;

long n;

char c;

fp = fopen("RANDOM", "w");

while((c = getchar()) != EOF)

putc(c,fp);

printf("No. of characters entered = %ld\n", ftell(fp));

fclose(fp);

fp = fopen("RANDOM","r");

n = 0L;

while(feof(fp) == 0)

{

fseek(fp, n, 0);

printf("Position of %c is %ld\n", getc(fp),ftell(fp));

n = n+5L;

}

putchar('\n');

fseek(fp,-1L,2);

do

{

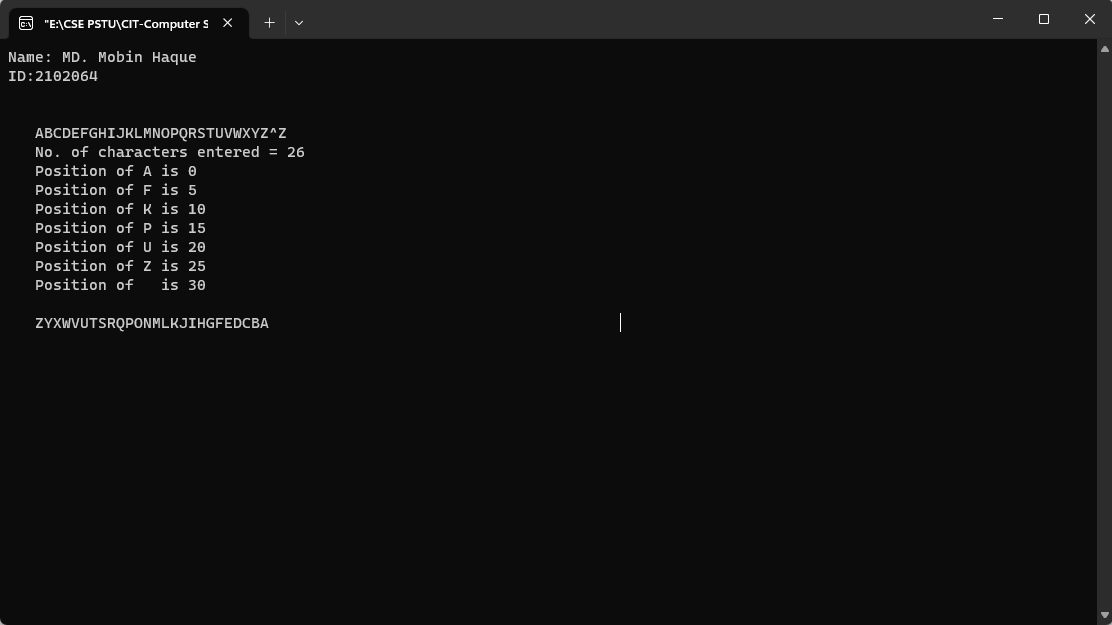
putchar(getc(fp));

}

while(!fseek(fp,-2L,1));

fclose(fp);

}



1. Write a program that will receive a filename and a line of text as command line arguments and write the text to the file.

#include<stdio.h>

main(argc, argv)

int argc;

char \*argv[];

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*fp;

int i;

char word[15];

fp = fopen(argv[1], "w");

printf("\nNo. of arguments in Command line = %d\n\n",argc);

for(i = 2; i < argc; i++)

fprintf(fp,"%s ", argv[i]);

fclose(fp);

printf("Contents of %s file\n\n", argv[1]);

fp = fopen(argv[1], "r");

for(i = 2; i < argc; i++)

{

fscanf(fp,"%s", word);

printf("%s ", word);

}

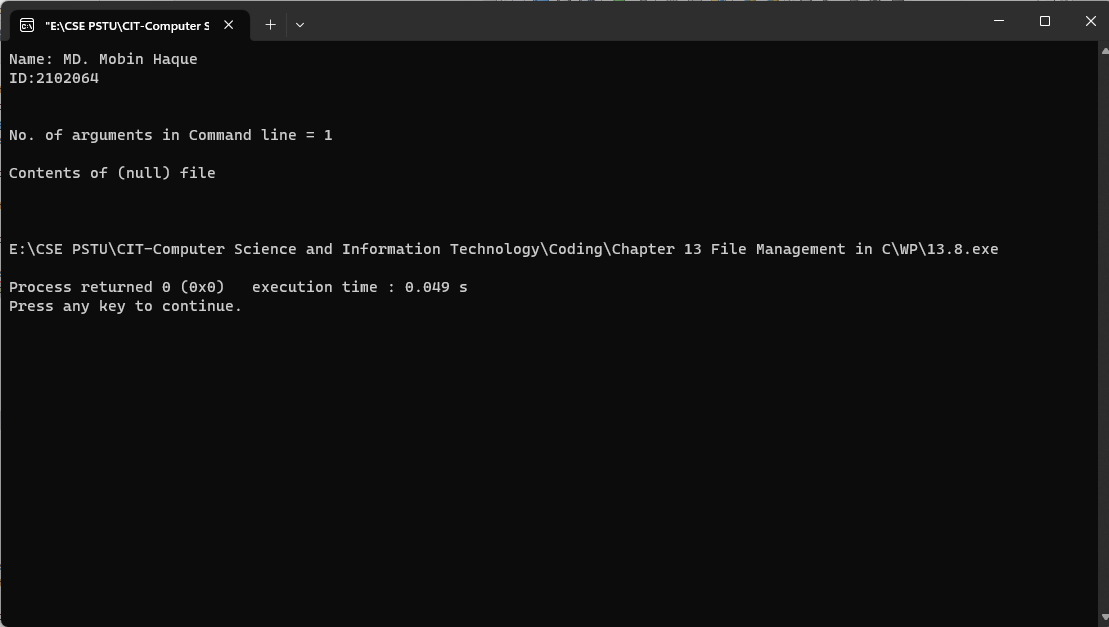
fclose(fp);

printf("\n\n");

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

printf("%\*s \n", i\*5,argv[i]);

}



1. Define a structure type, **struct personal** that would contain person name, date of joining and salary. Using this structure, write a program to read this information for one person from the keyboard and print the same on the screen.

struct personal

{

char name[20];

int day;

char month[10];

int year;

float salary;

};

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

struct personal person;

printf("Input Values\n");

scanf("%s %d %s %d %f",

person.name,

&person.day,

person.month,

&person.year,

&person.salary);

printf("%s %d %s %d %f\n",

person.name,

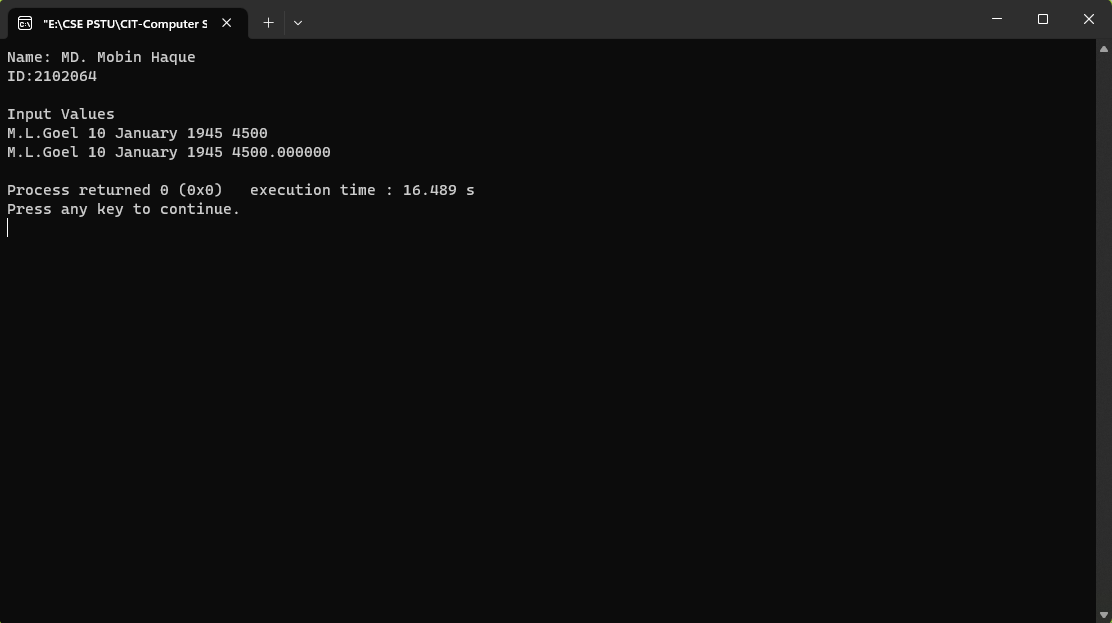
person.day,

person.month,

person.year,

person.salary);

}



1. Write a program to illustrate the comparison of structure variables.

struct class

{

int number;

char name[20];

float marks;

};

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

int x;

struct class student1 = {111,"Rao",72.50};

struct class student2 = {222,"Reddy", 67.00};

struct class student3;

student3 = student2;

x = ((student3.number == student2.number) &&

(student3.marks == student2.marks)) ? 1 : 0;

if(x == 1)

{

printf("\nstudent2 and student3 are same\n\n");

printf("%d %s %f\n", student3.number,

student3.name,

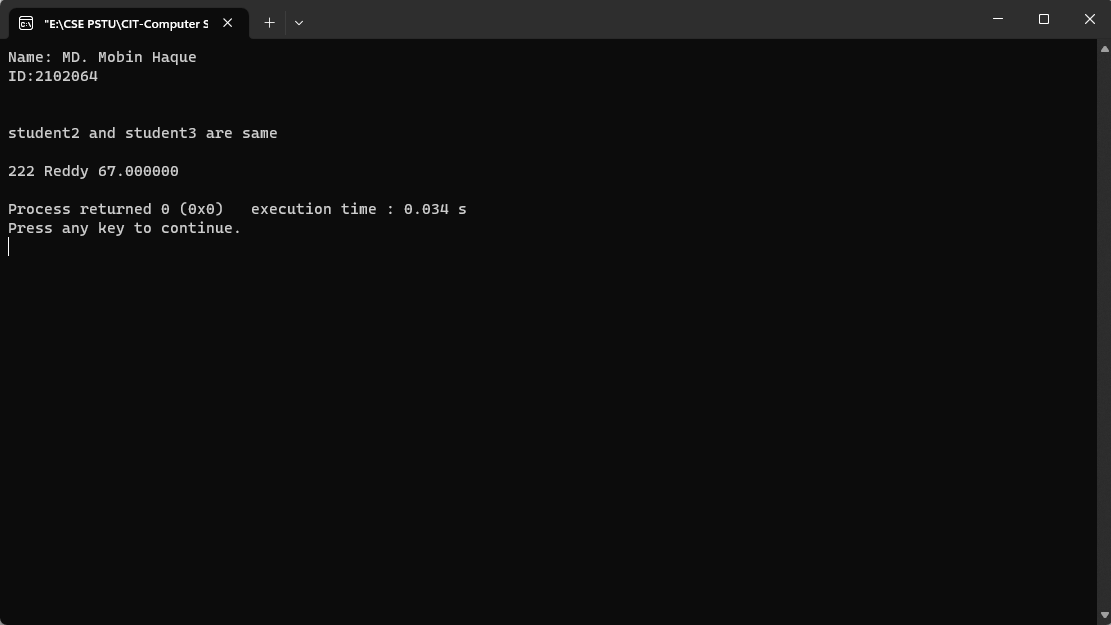
student3.marks);

}

else

printf("\nstudent2 and student3 are different\n\n");

}



1. For the **student** array discussed above, write a program to calculate the subject-wise and student-wise totals and store them as a part of the structure.

struct marks

{

int sub1;

int sub2;

int sub3;

int total;

};

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

int i;

struct marks student[3] = {{45,67,81,0},

{75,53,69,0},

{57,36,71,0}};

struct marks total;

for(i = 0; i <= 2; i++)

{

student[i].total = student[i].sub1 +

student[i].sub2 +

student[i].sub3;

total.sub1 = total.sub1 + student[i].sub1;

total.sub2 = total.sub2 + student[i].sub2;

total.sub3 = total.sub3 + student[i].sub3;

total.total = total.total + student[i].total;

}

printf(" STUDENT TOTAL\n\n");

for(i = 0; i <= 2; i++)

printf("Student[%d] %d\n", i+1,student[i].total);

printf("\n SUBJECT TOTAL\n\n");

printf("%s %d\n%s %d\n%s %d\n",

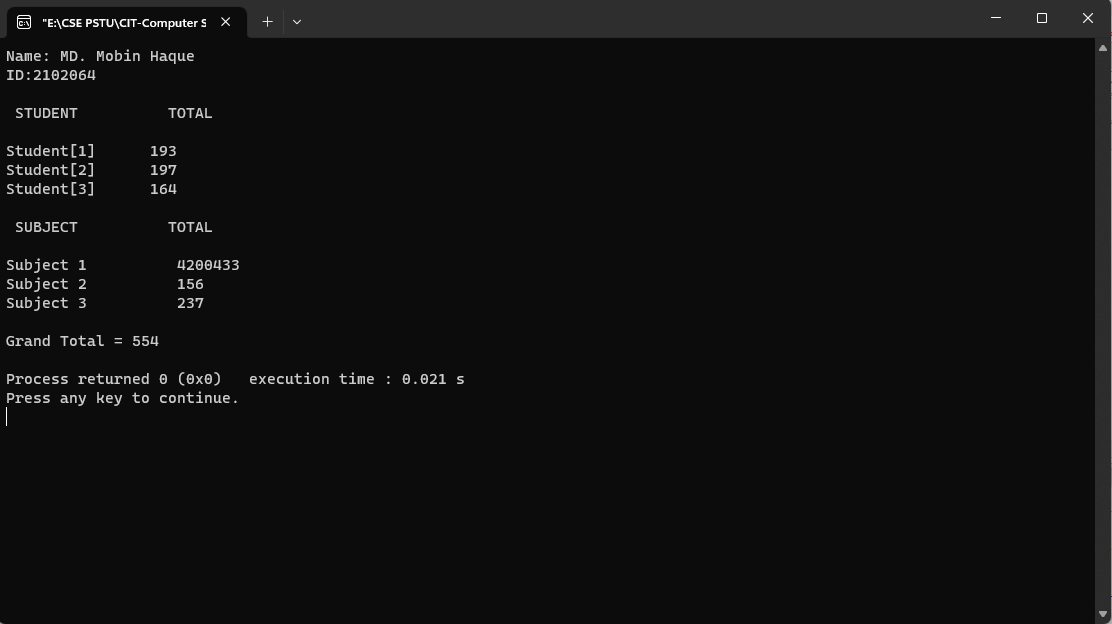
"Subject 1 ", total.sub1,

"Subject 2 ", total.sub2,

"Subject 3 ", total.sub3);

printf("\nGrand Total = %d\n", total.total);

}



1. Rewrite the program of Example 10.3 using an array member to represent the three subjects.

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

struct marks

{

int sub[3];

int total;

};

struct marks student[3] =

{45,67,81,0,75,53,69,0,57,36,71,0};

struct marks total;

int i,j;

for(i = 0; i <= 2; i++)

{

for(j = 0; j <= 2; j++)

{

student[i].total += student[i].sub[j];

total.sub[j] += student[i].sub[j];

}

total.total += student[i].total;

}

printf("STUDENT TOTAL\n\n");

for(i = 0; i <= 2; i++)

printf("Student[%d] %d\n", i+1, student[i].total);

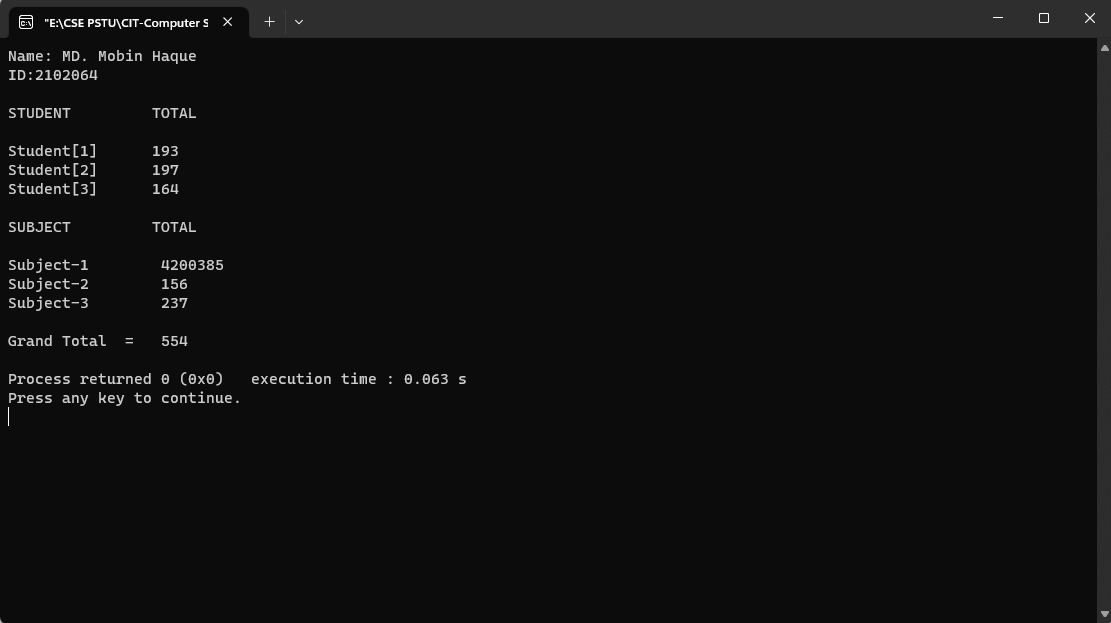
printf("\nSUBJECT TOTAL\n\n");

for(j = 0; j <= 2; j++)

printf("Subject-%d %d\n", j+1, total.sub[j]);

printf("\nGrand Total = %d\n", total.total);

}



1. Write a simple program to illustrate the method of sending an entire structure as parameter to a function.

struct stores

{

char name[20];

float price;

int quantity;

};

struct stores update (struct stores product, float p, int q);

float mul (struct stores stock);

main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

float p\_increment, value;

int q\_increment;

struct stores item = {"XYZ", 25.75, 12};

printf("\nInput increment values:");

printf(" price increment and quantity increment\n");

scanf("%f %d", &p\_increment, &q\_increment);

item = update(item, p\_increment, q\_increment);

printf("Updated values of item\n\n");

printf("Name : %s\n",item.name);

printf("Price : %f\n",item.price);

printf("Quantity : %d\n",item.quantity);

value = mul(item);

printf("\nValue of the item = %f\n", value);

}

struct stores update(struct stores product, float p, int q)

{

product.price += p;

product.quantity += q;

return(product);

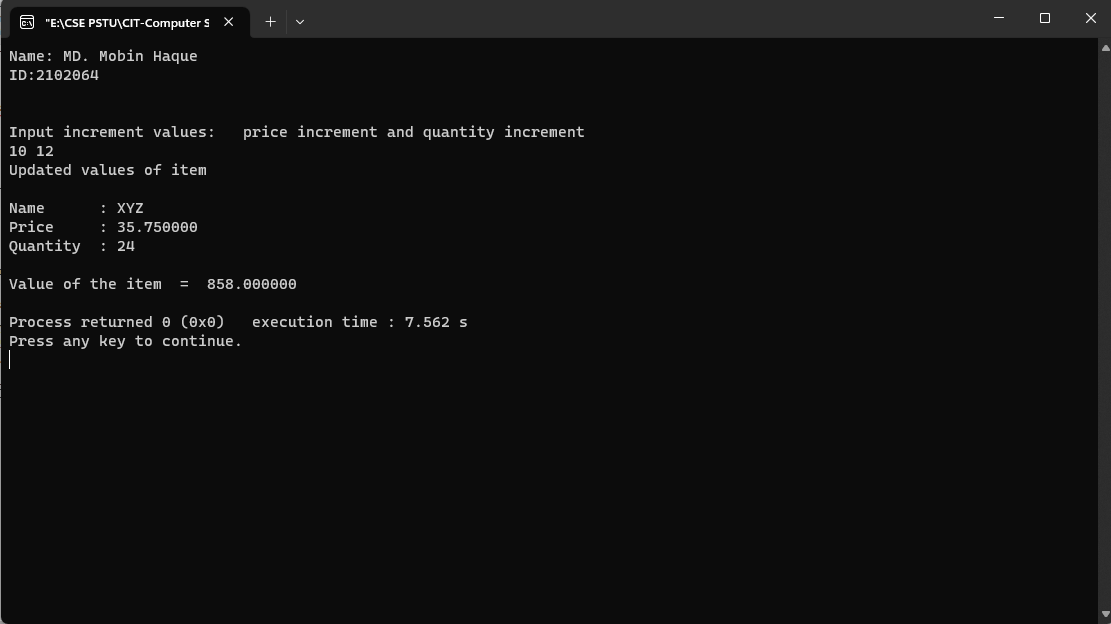
}

float mul(struct stores stock)

{

return(stock.price \* stock.quantity);

}



1. Write a program in C to create and store information in a text file.

#include<stdio.h>

#include<stdlib.h>

int main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

char str[1000];

FILE \*fptr;

char fname[20]="test.txt";

printf("\n\n Create a file (test.txt) and input text :\n");

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

fptr=fopen(fname,"w");

if(fptr == NULL)

{

printf(" Error in opening file!");

exit(1);

}

printf(" Input a sentence for the file : ");

fgets(str, sizeof str, stdin);

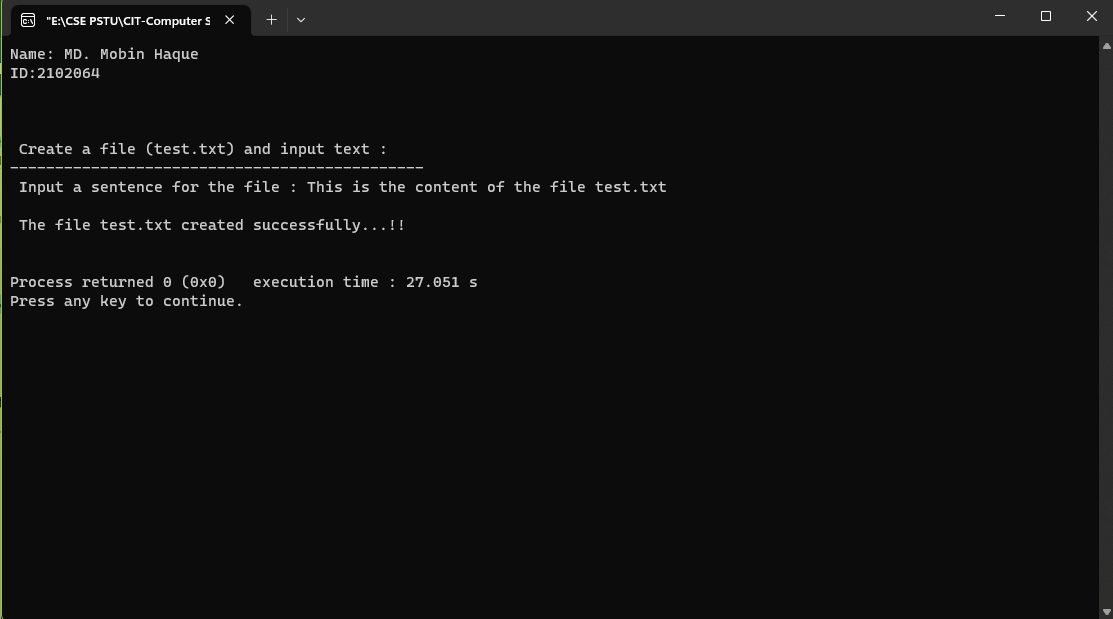
fprintf(fptr,"%s",str);

fclose(fptr);

printf("\n The file %s created successfully...!!\n\n",fname);

return 0;

}



1. Write a program in C to read an existing file.

#include<stdio.h>

#include<stdlib.h>

void main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \*fptr;

char fname[20];

char str;

printf("\n\n Read an existing file :\n");

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

printf(" Input the filename to be opened : ");

scanf("%s",fname);

fptr = fopen (fname, "r");

if (fptr == NULL)

{

printf(" File does not exist or cannot be opened.\n");

exit(0);

}

printf("\n The content of the file %s is :\n",fname);

str = fgetc(fptr);

while (str != EOF)

{

printf ("%c", str);

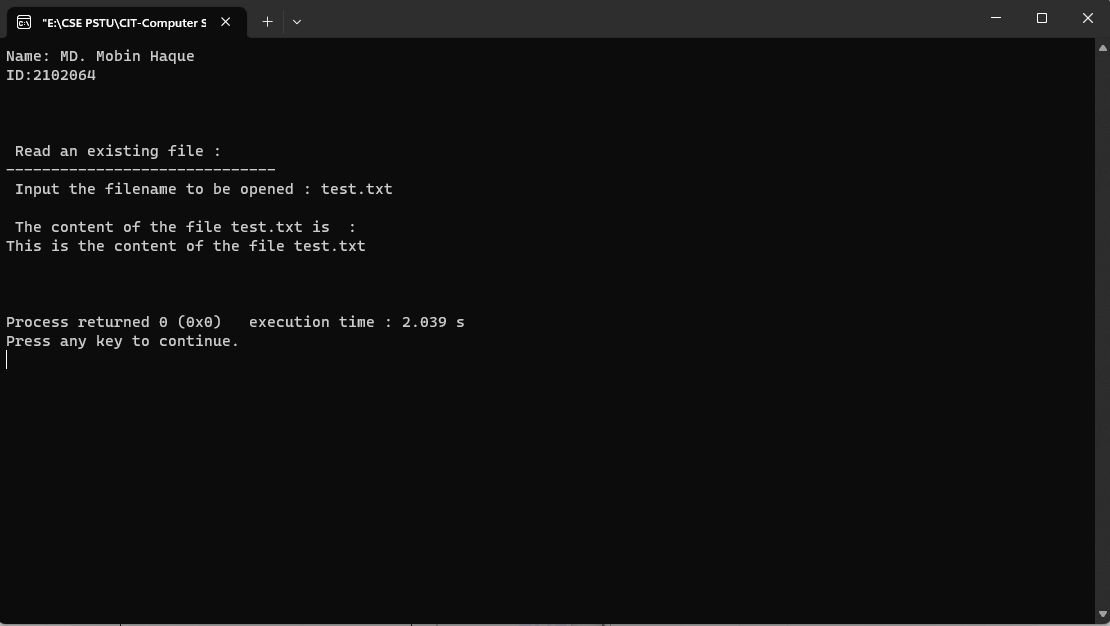
str = fgetc(fptr);

}

fclose(fptr);

printf("\n\n");

}



1. Write a program in C to write multiple lines to a text file.

#include<stdio.h>

int main()

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

FILE \* fptr;

int i,n;

char str[100];

char fname[20]="test.txt";

char str1;

printf("\n\n Write multiple lines in a text file and read the file :\n");

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

printf(" Input the number of lines to be written : ");

scanf("%d", &n);

printf("\n :: The lines are ::\n");

fptr = fopen (fname,"w");

for(i = 0; i < n+1;i++)

{

fgets(str, sizeof str, stdin);

fputs(str, fptr);

}

fclose (fptr);

/\*-------------- read the file -------------------------------------\*/

fptr = fopen (fname, "r");

printf("\n The content of the file %s is :\n",fname);

str1 = fgetc(fptr);

while (str1 != EOF)

{

printf ("%c", str1);

str1 = fgetc(fptr);

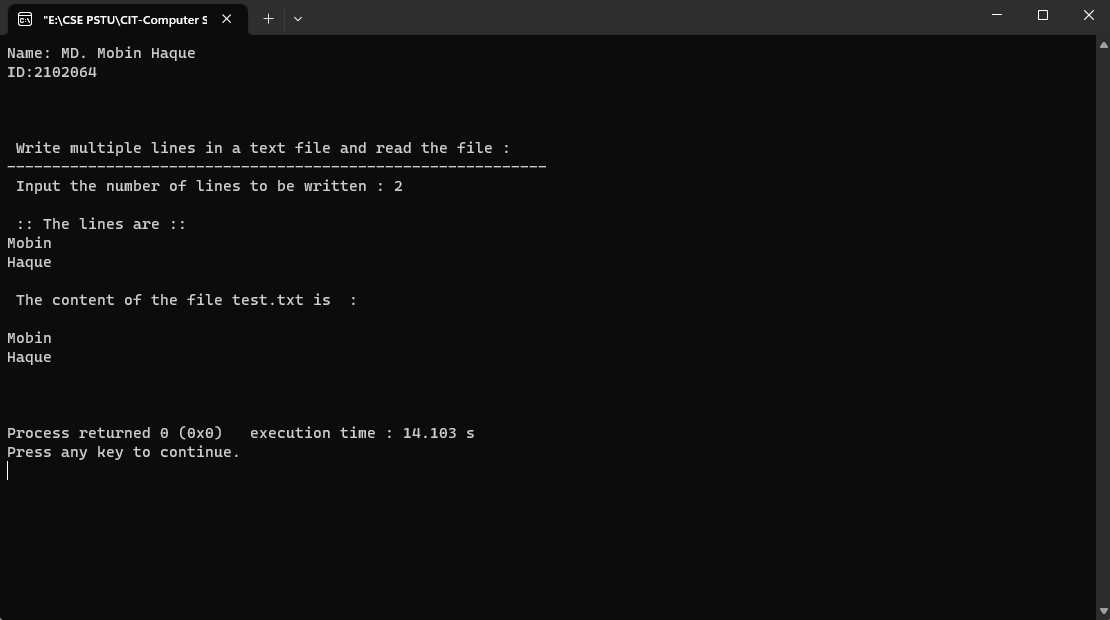
}

printf("\n\n");

fclose (fptr);

return 0;

}



1. Write a program in C to read the file and store the lines in an array.

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define LSIZ 128

#define RSIZ 10

int main(void)

{

printf("Name: Nazmus Sakib\nID:2102066\n\n");

char line[RSIZ][LSIZ];

char fname[20];

FILE \*fptr = NULL;

int i = 0;

int tot = 0;

printf("\n\n Read the file and store the lines into an array :\n");

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

printf(" Input the filename to be opened : ");

scanf("%s",fname);

fptr = fopen(fname, "r");

while(fgets(line[i], LSIZ, fptr))

{

line[i][strlen(line[i]) - 1] = '\0';

i++;

}

tot = i;

printf("\n The content of the file %s are : \n",fname);

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

{

printf(" %s\n", line[i]);

}

printf("\n");

return 0;

}

