QUESTION:1

**Write a C Program using Files for the following problem statements**

**1. Read and Write a specific character/string/Line from a Text File**

#include <stdio.h>

#include <fcntl.h>

#include <stdlib.h>

int to\_initcap\_file(FILE \*);

int main(int argc, char \* argv[])

{

FILE \*fp1;

char c[1000];

char fp[10];

int p;

fp1 = fopen("program.txt", "w");

if (fp1 == NULL) {

printf("Error!");

exit(1);

}

printf("Enter a sentence:\n");

fgets(c, sizeof(c), stdin);

fprintf(fp1, "%s", c);

fclose(fp1);

fp1 = fopen("program.txt", "r+");

if (fp1 == NULL)

{

printf("cannot open the file ");

exit(0);

}

p = to\_initcap\_file(fp1);

if (p == 1)

{

printf("success");

}

else

{

printf("failure");

}

fclose(fp1);

if ((fp1 = fopen("program.txt", "r")) == NULL) {

printf("Error! opening file");

exit(1);

}

fscanf(fp1, "%[^\n]", c);

printf("\nData from the file:\n%s", c);

fclose(fp1);

return 0;

}

int to\_initcap\_file(FILE \*fp)

{

return 1;

}

OUTPUT:

Enter a sentence:

Ram is boy.He goes to school everyday

success

Data from the file:

Ram is boy.He goes to school everyday

QUESTION:2

**2. Delete a specific character/string/Line from a Text File**

#include <stdio.h>

int main() {

FILE \*fp1, \*fp2;

char filename[40];

char c;

int del\_line, temp = 1;

printf("Enter file name: ");

scanf("%s", filename);

fp1 = fopen(filename, "r");

c = getc(fp1);

while (c != EOF)

{

printf("%c", c);

c = getc(fp1);

}

rewind(fp1);

printf(" \n Enter line number of the line to be deleted:");

scanf("%d", &del\_line);

fp2 = fopen("copy.c", "w");

c = getc(fp1);

while (c != EOF) {

c = getc(fp1);

if (c == '\n')

temp++;

if (temp != del\_line)

{

putc(c, fp2);

}

}

fclose(fp1);

fclose(fp2);

remove(filename);

rename("copy.c", filename);

printf("\n The contents of file after being modified are as follows:\n");

fp1 = fopen(filename, "r");

c = getc(fp1);

while (c != EOF) {

printf("%c", c);

c = getc(fp1);

}

fclose(fp1);

return 0;

}

OUTPUT:

Enter file name: program.txt

Ram is boy.He goes to school everyday

 Enter line number of the line to be deleted:1

 The contents of file after being modified are as  follows:

QUESTION:3

**3. Replace a specified character/string/Line in a Text File**

#include <stdio.h>

int main(void)

{

FILE \*fileptr1, \*fileptr2;

char filechar[40];

char c;

int delete\_line, temp = 1;

printf("Enter file name: ");

scanf("%s", filechar);

fileptr1 = fopen(filechar, "r");

c = getc(fileptr1);

while (c != EOF)

{

printf("%c", c);

c = getc(fileptr1);

}

printf(" \n Enter line number to be deleted and replaced");

scanf("%d", &delete\_line);

rewind(fileptr1);

fileptr2 = fopen("program.txt", "w");

c = getc(fileptr1);

while (c != EOF)

{

if (c == 'n')

{

temp++;

}

if (temp != delete\_line)

{

putc(c, fileptr2);

}

else

{

while ((c = getc(fileptr1)) != 'n')

{

}

printf("Enter new text");

fflush(stdin);

putc('n', fileptr2);

while ((c = getchar()) != 'n')

putc(c, fileptr2);

fputs("n", fileptr2);

temp++;

}

c = getc(fileptr1);

}

fclose(fileptr1);

fclose(fileptr2);

remove(filechar);

rename("replica.c", filechar);

fileptr1 = fopen(filechar, "r");

c = getc(fileptr1);

while (c != EOF)

{

printf("%c", c);

c = getc(fileptr1);

}

fclose(fileptr1);

return 0;

}

OUTPUT:

Enter file name:program.txt

hi.

hello

how are you?

hope the same

Enter line number of the line to be deleted and replaced:4

Enter new text: Meet u soon

hi.

hello

how are you?

Meet u soon

QUESTION:4

**4. Find the Number of character/string/Line in a Text File**

#include<stdio.h>

int main()

{

char str[200];

int line, word, ch;

line = word = ch = 0;

printf("Enter string terminated with ~ :\n");

scanf("%[^~]", str);

for(int i=0; str[i]!='\0'; i++)

{

if(str[i]=='\n')

{

line++;

word++;

}

else

{

if(str[i]==' '||str[i]=='\t')

{

word++;

ch++;

}

else {

ch++;

}

}

}

printf("\nCharacter counts = %d\n", ch);

printf("Word counts = %d\n", word);

printf("Line counts = %d\n", line);

return 0;

}

OUTPUT:

Enter string terminated with ~ :

Hello , how are u

welcome to the programming world

~

Character counts = 49

Word counts = 10

Line counts = 2

QUESTION:5

**5. Append the Content of File at the end of Another**

#include <stdio.h>

#include <stdlib.h>

main()

{

FILE \*fsring1, \*fsring2, \*ftemp;

char ch, file1[20], file2[20], file3[20];

printf("Enter name of first file ");

gets(file1);

printf("Enter name of second file ");

gets(file2);

printf("Enter name to store merged file ");

gets(file3);

fsring1 = fopen(file1, "r");

fsring2 = fopen(file2, "r");

if (fsring1 == NULL || fsring2 == NULL)

{

perror("Error has occured");

printf("Press any key to exit...\n");

exit(EXIT\_FAILURE);

}

ftemp = fopen(file3, "w");

if (ftemp == NULL)

{

perror("Error has occures");

printf("Press any key to exit...\n");

exit(EXIT\_FAILURE);

}

while ((ch = fgetc(fsring1)) != EOF)

fputc(ch, ftemp);

while ((ch = fgetc(fsring2) ) != EOF)

fputc(ch, ftemp);

printf("Two files merged %s successfully.\n", file3);

fclose(fsring1);

fclose(fsring2);

fclose(ftemp);

return 0;

}

OUTPUT:

Enter name of first file program.txt

Enter name of second file fname1

Enter name to store merged file mrg.txt

Two files merged  mrg.txt successfully.

QUESTION-6

**6. Merges Lines Alternatively from 2 Files & Print Result**

#include<stdio.h>

int main()

{

char file1[10], file2[10];

puts("enter the name of file 1");

scanf("%s", file1);

puts("enter the name of file 2");

scanf("%s", file2);

FILE \*fptr1, \*fptr2, \*fptr3;

fptr1=fopen(file1, "r");

fptr2=fopen(file2, "r");

fptr3=fopen("merge2.txt", "w+");

char str1[200];

char ch1, ch2;

int n = 0, w = 0;

while (((ch1=fgetc(fptr1)) != EOF) && ((ch2 = fgetc(fptr2)) != EOF))

{

if (ch1 != EOF)

{

ungetc(ch1, fptr1);

fgets(str1, 199, fptr1);

fputs(str1, fptr3);

if (str1[0] != 'n')

n+

}

if (ch2 != EOF)

{

ungetc(ch2, fptr2);

fgets(str1, 199, fptr2);

fputs(str1, fptr3);

if (str1[0] != 'n')

n++;

}

}

rewind(fptr3);

while ((ch1 = fgetc(fptr3)) != EOF)

{

ungetc(ch1, fptr3);

fscanf(fptr3, "%s", str1);

if (str1[0] != ' ' || str1[0] != 'n')

w++;

}

fprintf(fptr3, "\n\n number of lines = %d n number of words is = %d\n", n, w - 1);

fclose(fptr1);

fclose(fptr2);

fclose(fptr3);

}

OUTPUT:-

$ cc pgm51.c

$ a.out

enter the name of file 1

c.txt

enter the name of file 2

a.txt

$ vi merge2.txt

C is a general purpose procedural computer programming language .It support structure programming ,lexical variable scope and recursion with a static type system .By design, c provides constructs that map efficiently to typical machine instructions.it has found lasting use in applications previously coded in assembly language.

number of lines = 4

number of words is = 47

QUESTION-7

**7. Find Sum of Numbers given in Command Line Arguments**

#include <stdio.h>

int count, s = 0;

void sum(int \*, int \*);

void main(int argc , char \*argv[])

{

int i, ar[argc];

count = argc;

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

{

ar[i - 1] = atoi(argv[i]);

}

sum(ar, ar + 1);

printf("%d", s);

}

void sum(int \*a, int \* b)

{

if (count == 1)

return;

s = s + \*a + \*b;

count -= 2;

sum(a + 2, b + 2);

}

OUTPUT:-

$ cc arg4.c

$ a.out 1 2 3 4

sum is 10

QUESTION-8

**8. Capitalize First Letter of every Word in a File**

include <stdio.h>

#include <fcntl.h>

#include <stdlib.h>

int to\_initcap\_file(FILE \*);

void main(int argc, char \* argv[])

{

FILE \*fp1;

char fp[10];

int p;

fp1 = fopen(argv[1], "r+");

if (fp1 == NULL)

{

printf("cannot open the file ");

exit(0);

}

p = to\_initcap\_file(fp1);

if (p == 1)

{

printf("success");

}

else

{

printf("failure");

}

fclose(fp1);

}

int to\_initcap\_file(FILE \*fp)

{

char c;

c = fgetc(fp);

if (c >= 'a' && c <= 'z')

{

fseek(fp, -1L, 1);

fputc(c - 32, fp);

}

while(c != EOF)

{

if (c == ' ' || c == '\n')

{

c = fgetc(fp);

if (c >= 'a' && c <= 'z')

{

fseek(fp, -1L, 1);

fputc(c - 32, fp);

}

}

else

{

c = fgetc(fp);

}

}

return 1;

}

OUTPUT:-

$ cc file5.c

$ a.out sample

success

$ cat sample

Wipro Technologies

File Copy Function

Successfully Read

QUESTION NO 9

**9. Copy File into Another File**

#include <stdio.h>

void main(int argc,char \*\*argv)

{

FILE \*fp1, \*fp2;

char ch;

int pos;

if ((fp1 = fopen(argv[1],"r")) == NULL)

{

printf("\nFile cannot be opened");

return;

}

else

{

printf("\nFile opened for copy...\n ");

}

fp2 = fopen(argv[2], "w");

fseek(fp1, 0L, SEEK\_END);

pos = ftell(fp1);

fseek(fp1, 0L, SEEK\_SET);

while (pos--)

{

ch = fgetc(fp1);

fputc(ch, fp2);

}

fcloseall();

}

OUTPUT

$ cc fileex.c

$ a.out /tmp/vmlinux mylinux

File opened for copy...

$cmp /tmp/vmlinux mylinux

$ ls -l mylinux

-rw-rw-r--. 1 adi adi 3691146 Feb 14 19:57 mylinux

-------------------------------------------------------------------

QUESTION 10

**10.Convert the Content of File to LowerCase and UpperCase**

#include <stdio.h>

#include <errno.h>

int to\_lower\_file(FILE \*);

void main(int argc, char \* argv[])

{

int op = -1;

char ch;

FILE \*fp;

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

{

printf("FILE has been opened..!!!\n");

op = to\_lower\_file(fp);

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

fclose(fp);

}

Else

{

perror("Error Occured");

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

}

}

int to\_lower\_file(FILE \*f)

{

int c;

char ch;

while ((ch = fgetc(f))! = EOF)

{

c = (int)ch;

if (c >= 65 && c <= 90)

{

ch = ch + 32;

fseek(f, -1L, 1);

fputc(ch, f);

}

} return 0;

}

OUTPUT

$ gcc fileex.c

$ cat test1

THIS IS AN EXAMPLE

$ ./a.out test1

FILE has been opened..!!!

0

$ cat test1

this is an example

----------------------------------------

QUESTION:-11

**11.Convert the Content of File to UpperCase**

#include <stdio.h>

int to\_upper\_file(FILE \*);

int main(int argc,char \*argv[])

{

FILE \*fp;

int status;

if (argc == 1)

{

printf("Insufficient Arguments:");

printf("No File name is provided at command line");

return;

}

if (argc > 1)

{

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

status = to\_upper\_file(fp);

if (status == 0)

{

printf("\n The content of \"%s\" file was successfully converted to upper case\n",argv[1]);

return;

}

if (status == -1)

{

printf("\n Failed to convert");

return;

}

}

}

int to\_upper\_file(FILE \*fp)

{

char ch;

if (fp == NULL)

{

perror("Unable to open file");

return -1;

}

else

{

while (ch != EOF)

{

ch = fgetc(fp);

if ((ch >= 'a') && (ch <= 'z'))

{

ch = ch - 32;

fseek(fp,-1,SEEK\_CUR);

fputc(ch,fp);

}

}

return 0;

}

}

OUTPUT

/\* Input file : mydata

$ cat mydata

This is an example

\*/

$ gcc file3.c

$ a.out mydata

The content of "mydata" file was successfully converted to upper case

/\* "mydata" after conversion

$ cat mydata

THIS IS AN EXAMPLE

**QUESTION 12**

**12.Replace First Letter of every Word with Capital Letter**

#include <stdio.h>

#include <fcntl.h>

#include <stdlib.h>

int to\_initcap\_file(FILE \*);

int main(int argc, char \* argv[])

{

FILE \*fp1;

char c[1000];

char fp[10];

int p;

fp1 = fopen("program.txt", "w");

if (fp1 == NULL) {

printf("Error!");

exit(1);

}

printf("Enter a sentence:\n");

fgets(c, sizeof(c), stdin);

fprintf(fp1, "%s", c);

fclose(fp1);

fp1 = fopen("program.txt", "r+");

if (fp1 == NULL)

{

printf("cannot open the file ");

exit(0);

}

p = to\_initcap\_file(fp1);

if (p == 1)

{

printf("success");

}

else

{

printf("failure");

}

fclose(fp1);

if ((fp1 = fopen("program.txt", "r")) == NULL) {

printf("Error! opening file");

exit(1);

}

fscanf(fp1, "%[^\n]", c);

printf("\nData from the file:\n%s", c);

fclose(fp1);

return 0;

}

int to\_initcap\_file(FILE \*fp)

{

char c;

c = fgetc(fp);

if (c >= 'a' && c <= 'z')

{

fseek(fp, -1L, 1);

fputc(c - 32, fp);

}

while(c != EOF)

{

if (c == ' ' || c == '\n')

{

c = fgetc(fp);

if (c >= 'a' && c <= 'z')

{

fseek(fp, -1L, 1);

fputc(c - 32, fp);

}

}

else

{

c = fgetc(fp);

}

}

return 1;

}

OUTPUT

iter has some of the worst faculty i ever seen

success

Data from the file:

Iter Has Some Of The Worst Faculty I Ever Seen

**QUESTION 13**

**13.Count No of Lines, Blank Lines, Comments in a given Program**

#include <stdio.h>

void main(int argc, char\* argv[])

{

int line\_count = 0, n\_o\_c\_l = 0, n\_o\_n\_b\_l = 0, n\_o\_b\_l = 0, n\_e\_c = 0;

FILE \*fp1;

char ch;

fp1 = fopen("program.txt", "r");

while ((ch = fgetc(fp1))!= EOF)

{

if (ch == '\n')

{

line\_count++;

}

if (ch == '\n')

{

if ((ch = fgetc(fp1)) == '\n')

{

fseek(fp1, -1, 1);

n\_o\_b\_l++;

}

}

if (ch == ';')

{

if ((ch = fgetc(fp1)) == '\n')

{

fseek(fp1, -1, 1);

n\_e\_c++;

}

}

}

fseek(fp1, 0, 0);

while ((ch = fgetc(fp1))!= EOF)

{

if (ch == '/')

{

if ((ch = fgetc(fp1)) == '/')

{

n\_o\_c\_l++;

}

}

}

printf("Total no of lines: %d\n", line\_count);

printf("Total no of comment line: %d\n", n\_o\_c\_l);

printf("Total no of blank lines: %d\n", n\_o\_b\_l);

printf("Total no of non blank lines: %d\n", line\_count-n\_o\_b\_l);

printf("Total no of lines end with semicolon: %d\n", n\_e\_c);

}

OUTPUT

Total no of lines: 1

Total no of comment line: 0

Total no of blank lines: 0

Total no of non blank lines: 1

Total no of lines end with semicolon: 0

**QUESTION 14**

**14.Reverse the Contents of a File and Print it**

#include <stdio.h>

#include <errno.h>

long count\_characters(FILE \*);

long count\_characters(FILE \*f) ;

void main(int argc, char \* argv[])

{

int i;

long cnt;

char sentence[1000];

char ch, ch1;

FILE \*fp1, \*fp2, \*ptr;

if (fp1 = fopen("program.txt", "r"))

{

printf("The FILE has been opened...\n");

fp2 = fopen("prg", "w");

cnt = count\_characters(fp1); // to count the total number of characters inside the source file

fseek(fp1, -1L, 2); // makes the pointer fp1 to point at the last character of the file

printf("Number of characters to be copied %ld\n", ftell(fp1));

while (cnt)

{

ch = fgetc(fp1);

fputc(ch, fp2);

fseek(fp1, -2L, 1);

cnt--;

}

printf("\n\*\*new File creatd successfully by copying 1st file in reverse order\*\*\n");

}

else

{

perror("Error occured\n");

}

fclose(fp1);

fclose(fp2);

if ((ptr = fopen("prg", "r")) == NULL) {

printf("Error! opening file");

exit(1);

}

fscanf(ptr, "%[^\n]",sentence);

printf("\nData from the new file:\n%s", sentence);

fclose(ptr);

}

long count\_characters(FILE \*f)

{

fseek(f, -1L, 2);

long last\_pos = ftell(f);

return last\_pos;

}

**OUTPUT**

The FILE has been opened...

Number of characters to be copied 47

\*\*new File creatd successfully by copying 1st file in reverse order\*\*

Data from the new file:

  neeS revE I ytlucaF tsroW ehT fO emoS saH retI

tal no of lines: 1