**CA3001 – Programming and Data Structures using C**

**Assignment 13 - Lab 15**

**Q1. Read and Write a specific character/string/Line from a Text File**

**Ans – C Program:**

#include <stdio.h>

#include <stdlib.h>

int main()

{

char c[1000];

FILE \*fptr;

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

{

printf("Error! opening file");

exit(1);

}

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

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

fclose(fptr);

return 0;

}

**Q2. Delete a specific character/string/Line from a Text File**

**Ans – C Program:**

#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:abc.txt

hi.

Hello

how are you?

I am fine

hope the same

Enter line number of the line to be deleted:4

The contents of file after being modified are as follows:

hi.

hello

how are you?

hope the same

**Q3. Replace a specified character/string/Line in a Text File**

**Ans – C Program :**

#include <stdio.h>

#include <string.h>

#define MAX 256

int main()

{

FILE \*fptr1, \*fptr2;

int lno, linectr = 0;

char str[MAX],fname[MAX];

char newln[MAX], temp[] = "temp.txt";

printf(" Input the file name to be opened : ");

fgets(fname, MAX, stdin);

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

fptr1 = fopen(fname, "r");

if (!fptr1)

{

printf("Unable to open the input file!!\n");

return 0;

}

fptr2 = fopen(temp, "w");

if (!fptr2)

{

printf("Unable to open a temporary file to write!!\n");

fclose(fptr1);

return 0;

}

printf(" Input the content of the new line : ");

fgets(newln, MAX, stdin);

printf(" Input the line no you want to replace : ");

scanf("%d", &lno);

lno++;

while (!feof(fptr1))

{

strcpy(str, "\0");

fgets(str, MAX, fptr1);

if (!feof(fptr1))

{

linectr++;

if (linectr != lno)

{

fprintf(fptr2, "%s", str);

}

else

{

fprintf(fptr2, "%s", newln);

}

}

}

fclose(fptr1);

fclose(fptr2);

remove(fname);

rename(temp, fname);

printf(" Replacement did successfully..!! \n");

return 0;

}

**Output:**

Input the file name to be opened : test.txt

Input the content of the new line : 2

Input the line no you want to replace : 2

Replacement did successfully..!!

**Q4. Find the Number of character/string/Line in a Text File**

**Ans – C Program:**

#include <stdio.h>

int main()

{

FILE \*fp;

int no\_lines = 0;

char filename[40], sample\_chr;

printf("Enter file name: ");

scanf("%s", filename);

fp = fopen(filename, "r");

sample\_chr = getc(fp);

while (sample\_chr != EOF)

{

if (sample\_chr == '')

{

no\_lines=no\_lines+1;

}

sample\_chr = getc(fp);

}

fclose(fp);

printf("There are %d lines in %s ", no\_lines, filename);

return 0;

}

**Output:**

Enter file name:abc.txt

There are 4 lines in abc.txt

**Q5. Append the Content of File at the end of Another**

**Ans – C Program:**

#include <stdio.h>

void appendFiles(char source[],

char destination[])

{

FILE \*fp1, \*fp2;

fp1 = fopen(source, "a+");

fp2 = fopen(destination, "a+");

if (!fp1 && !fp2)

{

printf("Unable to open/" "detect file(s)\n");

return;

}

char buf[100];

fprintf(fp2, "\n");

while (!feof(fp1))

{

fgets(buf, sizeof(buf), fp1);

fprintf(fp2, "%s", buf);

}

rewind(fp2);

while (!feof(fp2))

{

fgets(buf, sizeof(buf), fp2);

printf("%s", buf);

}

}

int main()

{

char source[] = "file1.txt",

destination[] = "file2.txt";

appendFiles(source, destination);

return 0;

}

**Output:**

This is line one in file2

Programming is fun.

This is line one in file1

Hello World.

**Q6. Merges Lines Alternatively from 2 Files & Print Result**

**Ans – C Program:**

#include<stdio.h>

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);

{

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);

}

**Q7. Find Sum of Numbers given in Command Line Arguments**

**Ans – C Program:**

#include <stdio.h>

int count, s = 0;

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

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

{

int j, ar[argc];

count = argc;

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

{

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

}

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:**

0

**Q8. Capitalize First Letter of every Word in a File**

**Ans – C Program:**

#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(ch!= 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:**

Data Structures And Algorithm Concepts

**Q9. Copy File into Another File**

**Ans – C Program:**

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE \*fptr1, \*fptr2;

char filename[100], c;

printf("Enter the filename to open for reading \n");

scanf("%s", filename);

fptr1 = fopen(filename, "r");

if (fptr1 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

printf("Enter the filename to open for writing \n");

scanf("%s", filename);

fptr2 = fopen(filename, "w");

if (fptr2 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

c = fgetc(fptr1);

while (c != EOF)

{

fputc(c, fptr2);

c = fgetc(fptr1);

}

printf("\nContents copied to %s", filename);

fclose(fptr1);

fclose(fptr2);

return 0;

}

**Output:**

Enter the filename to open for reading

a.txt

Enter the filename to open for writing

b.txt

Contents copied to b.txt

**Q10.Convert the Content of File to LowerCase and UpperCase**

**Ans – C Program (Lower Case To Upper Case):**

#include <stdio.h>

int main()

{

FILE\* fptr;

char file[50] = { "gfg.txt" };

char ch;

fptr = fopen(file, "r");

ch = fgetc(fptr);

while (ch != EOF)

{

ch = toupper(ch);

printf("%c", ch);

ch = fgetc(fptr);

}

fclose(fptr);

return 0;

}

**Output:**

DATA STRUCTURES AND ALGORITHM CONCEPT

**Upper Case to Lower Case:**

#include <stdio.h>

int main()

{

FILE\* fptr;

char file[30] = { "gfg.txt" };

char ch;

fptr = fopen(file, "r");

ch = fgetc(fptr);

while (ch != EOF)

{

ch = tolower(ch);

printf("%c", ch);

ch = fgetc(fptr);

}

fclose(fptr);

return 0;

}

**Output:**

data structures and algorithm concepts

**Q11. Convert the Content of File to UpperCase**

**Ans – C Program:**

#include <stdio.h>

int main()

{

FILE\* fptr;

char file[50] = { "gfg.txt" };

char ch;

fptr = fopen(file, "r");

ch = fgetc(fptr);

while (ch != EOF)

{

ch = toupper(ch);

printf("%c", ch);

ch = fgetc(fptr);

}

fclose(fptr);

return 0;

}

**Output:**

DATA STRUCTURES AND ALGORITHM CONCEPT

**Q12.Replace First Letter of every Word with Capital Letter**

**Ans – C Program:**

#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(ch!= 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:**

Data Structures And Algorithm Concepts

**Q13.Count No of Lines, Blank Lines, Comments in a given Program**

**Ans – C 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(argv[1], "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: 204

Total no of comment line: 19

Total no of blank lines: 11

Total no of non blank lines: 193

Total no of lines end with semicolon: 66

**Q14.Reverse the Contents of a File and Print it**

**Ans – C Program:**

#include <stdio.h>

#include <string.h>

#define MAX 100

void reverseContent(char\* x)

{

FILE\* fp = fopen(x, "a+");

if (fp == NULL)

{

printf("Unable to open file\n");

return;

}

char buf[100];

int a[MAX], s = 0, c = 0, l;

fprintf(fp, " \n");

rewind(fp);

while (!feof(fp))

{

fgets(buf, sizeof(buf), fp);

l = strlen(buf);

a = s += l;

}

rewind(fp);

c -= 1;

while (c >= 0)

{

fseek(fp, a, 0);

fgets(buf, sizeof(buf), fp);

printf("%s", buf);

c--;

}

return ;

}

int main()

{

char x[] = "file1.txt";

reverseContent(x);

return 0;

}

**Output:**

Input:

file1.txt has:

This is line one

This is line two

This is line three

This is line four

This is line five

Output:

This is line five

This is line four

This is line three

This is line two

This is line one