# CENTRAL CALCUTTA POLYTECHNIC

21, Convent Road, Philips, Sealdah, Kolkata, West Bengal 700014

DEPT. : COMPUTER SCIENCE AND TECHNOLOGY

- NAME : SUMAN MONDAL
- ROLL : DCCPCSTS6
- NUMBER : 10005537
- REG NUMBER : D192005242
- SUBJECT : SYSTEM PROGRAMMING AND COMPILER DESIGN
- SESSION : 2021 - 2022
- EMAIL : SUMAN.MONDAL@OUTLOOK.IN

# Contents

# Chapter 1

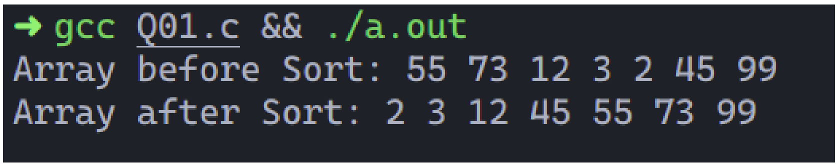
# SPCD Assignment-2

## 1.1 Implement Radix sort using C language

Source Code :

```c
//Implement Radix sort using C language

#include <stdio.h>

// Function to get the largest element from an array
int getMax(int array[], int n) {
  int max = array[0];
  for (int i = 1; i < n; i++)
    if (array[i] > max)
      max = array[i];
  return max;
}

// Using counting sort to sort the elements in the basis of
 ↪  significant places
void countingSort(int array[], int size, int place) {
  int output[size + 1];
  int max = (array[0] / place) % 10;

  for (int i = 1; i < size; i++) {
    if (((array[i] / place) % 10) > max)
      max = array[i];
  }
  int count[max + 1];

  for (int i = 0; i < max; ++i)
    count[i] = 0;

  // Calculate count of elements
  for (int i = 0; i < size; i++)
    count[(array[i] / place) % 10]++;

  // Calculate cumulative count
  for (int i = 1; i < 10; i++)
    count[i] += count[i - 1];

  // Place the elements in sorted order
  for (int i = size - 1; i >= 0; i--) {
    output[count[(array[i] / place) % 10] - 1] = array[i];
    count[(array[i] / place) % 10]--;
  }

  for (int i = 0; i < size; i++)
    array[i] = output[i];
}

// Main function to implement radix sort
void radixsort(int array[], int size) {
  // Get maximum element
  int max = getMax(array, size);

  // Apply counting sort to sort elements based on place value.
  for (int place = 1; max / place > 0; place *= 10)
    countingSort(array, size, place);
}

// Print an array
void printArray(int array[], int size) {
  for (int i = 0; i < size; ++i) {
    printf("%d ", array[i]);
  }
  printf("\n");
}

// Driver code
int main() {
  int array[] = {55, 73, 12, 3, 2, 45, 99};
  printf("Array before Sort: ");
  for (int k = 0; k < 7; k++)
    printf ("%d ", array[k]);
  printf ("\n");
  int n = sizeof(array) / sizeof(array[0]);
  radixsort(array, n);
  printf("Array after Sort: ");
  printArray(array, n);
}
```

Program Output :



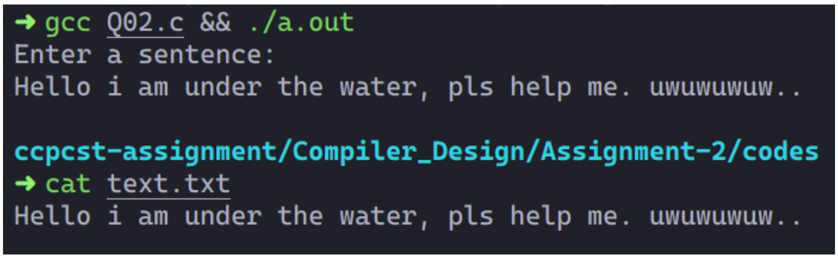
```
➜ gcc Q01.c && ./a.out
Array before Sort: 55 73 12 3 2 45 99
Array after Sort: 2 3 12 45 55 73 99
```

### 1.2 C Program to write text in a file

Source Code :

```
// C Program to write text in a file.

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

int main() {
    char sentence[1000];

    // creating file pointer to work with files
    FILE *fptr;

    // opening file in writing mode
    fptr = fopen("text.txt", "w");

    // exiting program
    if (fptr == NULL) {
        printf("Error!");
        exit(1);
    }
    printf("Enter a sentence:\n");
    fgets(sentence, sizeof(sentence), stdin);
    fprintf(fptr, "%s", sentence);
    fclose(fptr);
    return 0;
}
```

Program Output :



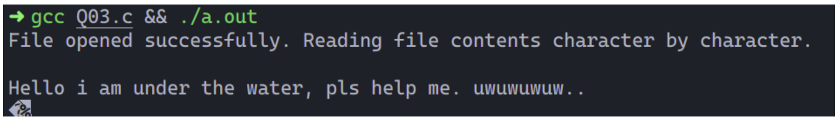
```
➜ gcc Q02.c && ./a.out
Enter a sentence:
Hello i am under the water, pls help me. uwuwuwuw..

ccpcst-assignment/Compiler_Design/Assignment-2/codes
➜ cat text.txt
Hello i am under the water, pls help me. uwuwuwuw..
```

### 1.3 C program to read a file and display file contents

Source Code :

```c
//  C program to read a file and display file contents


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


int main()
{
    /* File pointer to hold reference to our file */
    FILE * fPtr;

    char ch;


    /*
     * Open file in r (read) mode.
     * "data/file1.txt" is complete file path to read
     */
    fPtr = fopen("text.txt", "r");


    /* fopen() return NULL if last operation was unsuccessful */
    if(fPtr == NULL)
    {
        /* Unable to open file hence exit */
        printf("Unable to open file.\n");
        printf("Please check whether file exists and you have read
         ↪  privilege.\n");
        exit(EXIT_FAILURE);
    }
    /* File open success message */
    printf("File opened successfully. Reading file contents
     ↪  character by character. \n\n");

    do
    {
        /* Read single character from file */
        ch = fgetc(fPtr);

        /* Print character read on console */
        putchar(ch);

    } while(ch != EOF); /* Repeat this if last read character is
     ↪  not EOF */


    /* Done with this file, close file to release resource */
    fclose(fPtr);


    return 0;
}
```

Program Output :



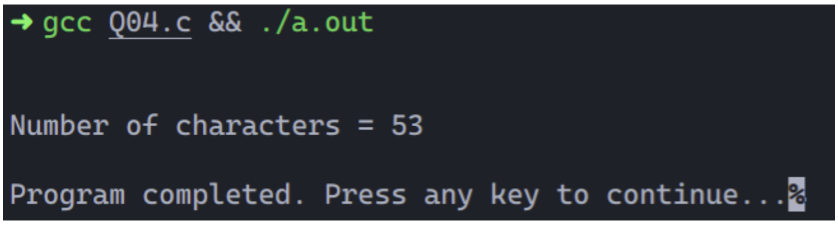
```
➜ gcc Q03.c && ./a.out
File opened successfully. Reading file contents character by character.

Hello i am under the water, pls help me. uwuwuwuw..
```

## 1.4 C Program to count number of characters in a text file

Source Code :

```
// C Program to count number of characters in a text file

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

int main()
{
    FILE *fptr;
    char ch;
    int count=0;
    /* Opening file in read mode */
    fptr = fopen("text.txt","r");
    if(fptr==NULL)
    {
        printf("Can't open file. Make sure file exits.\n");
        exit(1);
    }

    /* Counting characters */
    do
    {
        ch = fgetc(fptr);
        count++;
    }while(ch!=EOF);

    fclose(fptr);
    printf("\n\nNumber of characters = %d",count);
    printf("\n\nProgram completed. Press any key to continue...");

    return 0;
}
```

Program Output :



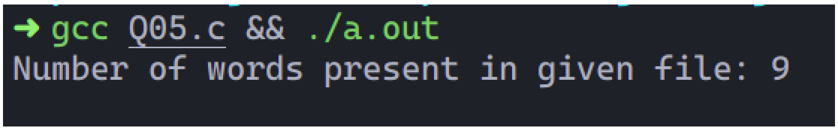
```
➜ gcc Q04.c && ./a.out


Number of characters = 53

Program completed. Press any key to continue...%
```

### 1.5 C Program to Count Number of words in a text file

Source Code :

```
// C Program to Count Number of words in a text file

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

int main()
{   char ch;
    FILE *file;
    int count = 0;
    //Opens a file in read mode
    file = fopen("text.txt","r");

    //Gets each character till end of file is reached
    while((ch = fgetc(file)) != EOF){
        //Counts each word
        if(ch ==' ' || ch == '\n')
            count++;
    }

    printf("Number of words present in given file: %d\n", count);
    fclose(file);

    return 0;
}
```

Program Output :



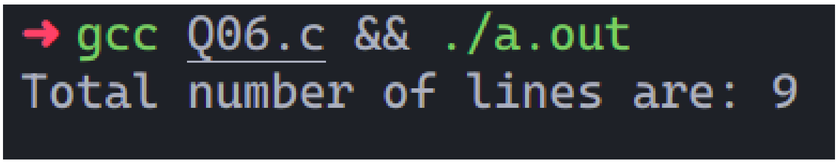
```
➜ gcc Q05.c && ./a.out
Number of words present in given file: 9
```

### 1.6 C Program to count number of lines in a text file

Source Code :

```
// C Program to count number of lines in a text file

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

#define FILENAME "text.txt"
int main() {
   FILE *fp;
   char ch;
   int linesCount=0;
   //open file in read more
   fp=fopen(FILENAME,"r");
   if(fp==NULL) {
      printf("File \"%s\" does not exist!!!\n",FILENAME);
      return -1;
   }
   //read character by character and check for new line
   while((ch=fgetc(fp))!=EOF) {
      if(ch=='\n')
         linesCount++;
   }
   //close the file
   fclose(fp);
   //print number of lines
   printf("Total number of lines are: %d\n",linesCount);
   return 0;
}
```

Program Output :



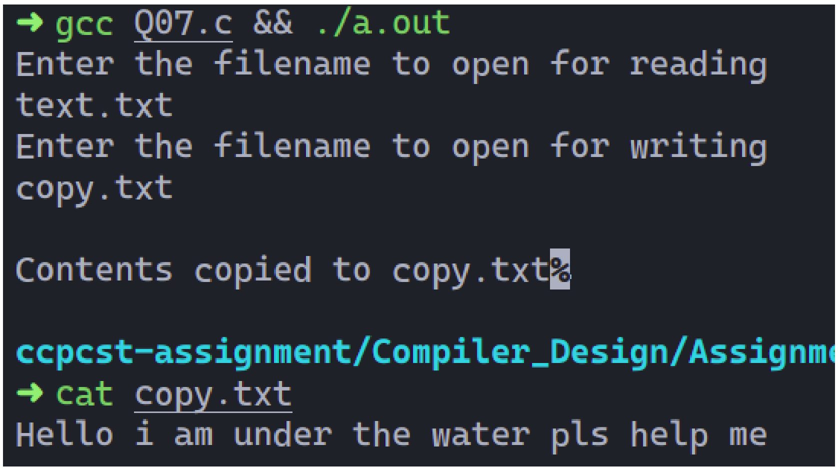
```
➜ gcc Q06.c && ./a.out
Total number of lines are: 9
```

### 1.7 C Program to copy contents of one file to another file

Source Code :

```
// C Program to copy contents of one file to another file

#include <stdio.h>
#include <stdlib.h> // For exit()
int main(){
    FILE *fptr1, *fptr2;
    char filename[100], c;
    printf("Enter the filename to open for reading \n");
    scanf("%s",filename);
    // Open one file for reading
    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);
    // Open another file for writing
    fptr2 = fopen(filename, "w");
    if (fptr2 == NULL){
        printf("Cannot open file %s \n", filename);
        exit(0);
    }
    // Read contents from file
    c = fgetc(fptr1);
    while (c != EOF){
        fputc(c, fptr2);
        c = fgetc(fptr1);
    }
    printf("\nContents copied to %s", filename);
    fclose(fptr1);
    fclose(fptr2);
    return 0;
}
```

## Program Output :



```
➜ gcc Q07.c && ./a.out
Enter the filename to open for reading
text.txt
Enter the filename to open for writing
copy.txt

Contents copied to copy.txt%

ccpcst-assignment/Compiler_Design/Assignme
➜ cat copy.txt
Hello i am under the water pls help me
```

### 1.8 Write an algorithm to implement Lexical Analyzer

*Algorithm :*

```
// Write an algorithm to implement Lexical Analyzer

Step 1 : Start the program
Step 2 : Include necessary header files.
Step 3: The ctype header file is to load the file with predicate isdigit.
Step 4 : The define directive defines the buffer size, numerics, assignment
 ↪  operator, relational operator.
Step 5 : Initialize the necessary variables.
Step 6: To return index of new string S, token t using insert() function.
Step 7 : Initialize the length of every string.
Step 8: Check the necessary condition.
Step 9: Call the initialize() function. This function loads the keywords into the
 ↪  symbol table.
Step 10 : Check the conditions such as white spaces, digits, letters and
 ↪  alphanumerics.
Step 11 : To return index of entry for string S, or 0 if S is not found using
 ↪  lookup( ) function.
Step 12 : Check this until EOF is found.
Step 13 : Otherwise initialize the token value to be none.
Step 14 : In the main function if lookahead equals numeric then the value of
 ↪  attribute num is given by the global variable tokenval.
Step 15 : Check the necessary conditions such as arithmetic operators ,
 ↪  parenthesis , identifiers, assignment operators and relational operators.
Step 16 : Stop the program.
```