1. main.c

```
#include <stdio.h>
#include "header.h"
#include <string.h>
#include <stdlib.h>
#include <wchar.h>
int main() {
   int choice;
   printf("Enter Question Number (1-15): ");
   scanf("%d", &choice);
   /*
      Clear the input buffer to handle the newline character after scanf("%d", &question)
   */
   getchar();
   switch (choice) {
      case 1: {
         char str[100], c;
         printf("Enter a string: ");
         scanf("%[^\n]s", str);
         getchar();
         printf("Enter a character: ");
         scanf(" %c", &c);
         int count = countOccurrences(str, c);
         printf("The character '%c' occurs %d times.\n", c, count);
         break;
      }
      case 2: {
         char str[100], *token;
         printf("Enter a string: ");
         scanf("%[^\n]s", str);
         token = my_strtok(str, " ,?");
```

```
    while (token) {
        printf("%s\n", token);
        token = my_strtok(NULL, " ,?");
    }
    break;
}
case 3: {
    char str1[100], str2[100], *subseq;
    printf("Enter first string: ");
    scanf("%[^\n]s", str1);
    getchar();
    printf("Enter second string: ");
    scanf("%[^\n]s", str2);
    int len = longestSubsequence(str1, str2, &subseq);
    if (len > 0) {
        printf("Longest subsequence: %.*s\n", len, subseq);
    } else {
        printf("No common subsequence found.\n");
    }
    break;
}
case 4: {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
    int result = gcd(a, b);
    printf("GCD of %d and %d is %d.\n", a, b, result);
    break;
}
case 5: {
    int a, b;
```

```
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
    int result = lcm(a, b);
    printf("LCM of %d and %d is %d.\n", a, b, result);
    break;
}
case 6: {
    int n;
    char binary[33];
    printf("Enter a decimal number: ");
    scanf("%d", &n);
    printf("Binary representation: %s\n", decimalToBinary(n, binary));
    break;
}
case 7: {
    char str1[100], str2[100];
    printf("Enter first string: ");
    scanf("%[^\n]s", str1);
    getchar();
    printf("Enter second string: ");
    scanf("%[^\n]s", str2);
    int result = my_strcasecmp(str1, str2);
    if (result == 0) {
        printf("The strings are equal.\n");
    } else {
        printf("The strings are not equal.\n");
    }
    break;
}
case 8: {
    char str[100];
```

```
    char *word;
    char *str_ptr;
    char temp[100];

    printf("Enter a string: ");
    scanf(" %[^\n]s", temp);
    str_ptr = temp;
    strncpy(str, temp, sizeof(str) - 1);
    str[sizeof(str) - 1] = '\0';

    word = my_strsep(&str_ptr, " ");
    while (word != NULL) {
        printf("Word: %s\n", word);
        word = my_strsep(&str_ptr, " ");
    }
    break;
}
case 9: {
    char str1[100], str2[100];
    printf("Enter first string: ");
    scanf("%[^\n]s", str1);
    getchar();
    printf("Enter second string: ");
    scanf("%[^\n]s", str2);
    int result = my_strcoll(str1, str2);
    if (result == 0) {
        printf("The strings are equal.\n");
    } else if (result < 0) {
        printf("String 1 is less than String 2.\n");
    } else {
        printf("String 1 is greater than String 2.\n");
```

```
    }
    break;
}
case 10:{
    wchar_t str1[100];
    wchar_t str2[100];
    wprintf(L"Enter the first wide-character string: ");
    wscanf(L"%ls", str1);

    wprintf(L"Enter the second wide-character string: ");
    wscanf(L"%ls", str2);
    int result = wcscmp(str1, str2);
    if (result == 0) {
        wprintf(L"The two wide-character strings are equal.\n");
    } else if (result < 0) {
        wprintf(L"The first wide-character string is less than the second.\n");
    } else {
        wprintf(L"The first wide-character string is greater than the second.\n");
    }
    break;
}
case 11:{
    double x, result_sin, result_asin, result_cos, result_acos, result_tan, verify;
    printf("Enter a value for x (in radians): ");
    scanf("%lf", &x);
    result_sin = my_sin(x);
    result_asin = my_asin(result_sin);
    result_cos = my_cos(x);
    result_acos = my_acos(result_cos);
    result_tan = my_tan(x);
    verify = my_sin(my_asin(x));
```

```
    printf("sin(%.2lf) = %.6lf\n", x, result_sin);
    printf("asin(sin(%.2lf)) = %.6lf\n", x, result_asin);
    printf("cos(%.2lf) = %.6lf\n", x, result_cos);
    printf("acos(cos(%.2lf)) = %.6lf\n", x, result_acos);
    printf("tan(%.2lf) = %.6lf\n", x, result_tan);
    printf("Verification: sin(asin(%.2lf)) = %.6lf\n", x, verify);
    break;
}
case 12:{
    int number;
    printf("Enter a number: ");
    scanf("%d", &number);
    int reverse = reverseInteger(number);
    printf("The reversed integer is: %d", reverse);
    break;
}
case 13:{
    char str[100];
    printf("Enter a string: ");
    scanf("%s", str);
    if(isAllDigits(str)){
        int number = atoi(str);
        printf("The integer representation is: %d\n", number);
    }else{
        printf("The string contains non-digit characters.\n");
    }
    break;
}
case 14: {
    char str[100];
    printf("Enter a string: ");
```

```
            scanf("%[^\n]s", str);
            rev(str);
            printf("Reversed string: %s\n", str);
            break;
        }
        case 15: {
            char str[100], ch;
            printf("Enter a string: ");
            scanf("%[^\n]s", str);
            printf("Enter a character: ");
            scanf(" %c", &ch);
            char *result = cutonchar(str, ch);
            printf("Result after cut: %s\n", result);
            break;
        }
        default:
            printf("Invalid choice.\n");
            break;
    }
    return 0;
}
```

2. header.h

```
int countOccurrences(char* str, char ch);
char *my_strtok(char *str, const char *delim);
int longestSubsequence(char *str1, char *str2, char **subseq);
int gcd(int a, int b);
int lcm(int a, int b);
char *decimalToBinary(int n, char *binary);
int my_strcasecmp(const char *s1, const char *s2);
char *my_strsep(char **stringp, const char *delim);
int my_strcoll(const char *s1, const char *s2);
double my_sin(double x);
double my_cos(double x);
double my_asin(double x);
double my_acos(double x);
double my_tan(double x);
int reverseInteger(int number);
int isAllDigits(const char *str);
void rev(char *str);
char *cutonchar(char *str, char ch);
```

3. logic.c

```
#include <stdio.h>
#include "header.h"
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#include <math.h>
int countOccurrences(char *str, char ch) {
    int count = 0;
    ch = tolower(ch);
    while (*str) {
        if (tolower(*str) == ch) {
            count++;
        }
        str++;
    }
    return count;
}

char *my_strtok(char *str, const char *delim){
    static char *nextToken = NULL;
    if(str != NULL){
        nextToken = str;
    }
    if(nextToken == NULL){
        return NULL;
    }
    char *tokenStart = nextToken;
    while(*nextToken){
        const char *d = delim;
        while(*d){
```

```
            if(*nextToken == *d){
                *nextToken = '\0';
                nextToken++;
                while(*nextToken && strchr(delim, *nextToken)){
                    nextToken++;
                }
                return tokenStart;
            }
            d++;
        }
        nextToken++;
    }
    nextToken = NULL;
    return tokenStart;
}
int longestSubsequence(char *str1, char *str2, char **subseq) {
    int maxLen = 0;
    char *bestMatch = NULL;

    for (int i = 0; str2[i] != '\0'; i++) {
        for (int j = 0; str1[j] != '\0'; j++) {
            int len = 0;
            while (str1[j + len] && str2[i + len] && str1[j + len] == str2[i + len]) len++;
            if (len > maxLen) {
                maxLen = len;
                bestMatch = &str1[j];
            }
        }
    }
    *subseq = bestMatch;
    return maxLen;
```

```
}

int gcd(int a, int b) {
   if (b == 0) return a;
   return gcd(b, a % b);
}

int lcm(int a, int b) {
   return (a * b) / gcd(a, b);
}

char *decimalToBinary(int n, char *binary) {
   int i = 0;
   while (n > 0) {
      binary[i++] = (n % 2) + '0';
      n /= 2;
   }
   binary[i] = '\0';
   rev(binary);
   return binary;
}

int my_strcasecmp(const char *s1, const char *s2) {
   while (*s1 && (tolower(*s1) == tolower(*s2))) {
      s1++;
      s2++;
   }
   return tolower(*(unsigned char *)s1) - tolower(*(unsigned char *)s2);
}
char *my_strsep(char **stringp, const char *delim) {
   char *start = *stringp;
```

```
    char *p;
    if (!start) return NULL;

    p = strpbrk(start, delim);
    if (p) {
        *p = '\0';
        *stringp = p + 1;
    } else {
        *stringp = NULL;
    }

    return start;
}
int my_strcoll(const char *s1, const char *s2) {
    return strcmp(s1, s2);
}
double my_sin(double x){
    return sin(x);
}
double my_asin(double x){
    return asin(x);
}
double my_cos(double x){
    return cos(x);
}
double my_acos(double x){
    return acos(x);
}
double my_tan(double x){
    return my_sin(x) / my_cos(x);
}
```

```
int reverseInteger(int number){
   int reverse = 0;
   while (number != 0) {
      int remainder = number % 10;
      reverse = reverse * 10 + remainder;
      number = number / 10;
   }
   return reverse;
}

int isAllDigits(const char *str){
   int i = 0;
   while (str[i] != '\0') {
      if (!isdigit(str[i])) {
         return 0;
      }
      i++;
   }
   return 1;
}

void rev(char *str) {
   int len = strlen(str);
   for (int i = 0; i < len / 2; i++) {
      char temp = str[i];
      str[i] = str[len - 1 - i];
      str[len - 1 - i] = temp;
   }
}
```

```
char *cutonchar(char *str, char ch) {
    char *pos = strchr(str, ch);
    if(pos){
        *pos = '\0';
    }
    return str;
}
```

Outputs of String Assignment:

```
E:\COEP\DSA\Assignments\StringLabAssignment>gcc -Wall main.c logic.c -o a

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 1
Enter a string: Tanishq Ganesh Tote
Enter a character: T
The character 'T' occurs 3 times.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 2
Enter a string: Hello, How are you?
Hello
How
are
you


E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 3
Enter first string: Tanishq Tote
Enter second string: Aishwarya Tote
Longest subsequence:  Tote

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 4
Enter two numbers: 35 49
GCD of 35 and 49 is 7.
```

```
Enter Question Number (1-15): 5
Enter two numbers: 22 23
LCM of 22 and 23 is 506.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 6
Enter a decimal number: 23
Binary representation: 10111

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 7
Enter first string: abcde fghij klmno pqrstu
Enter second string: abcdefghijklmnopqrstu
The strings are not equal.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 7
Enter first string: abcde fghij klmno pqrst
Enter second string: abcde fghij klmno pqrst
The strings are equal.
```

```
E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 8
Enter a string: I am pursuing BTech in Computer Science
Word: I
Word: am
Word: pursuing
Word: BTech
Word: in
Word: Computer
Word: Science

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 9
Enter first string: Hey I am Tanishq
Enter second string: Hey I am Tanishq
The strings are equal.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 10
Enter the first wide-character string: Hello
Enter the second wide-character string: hello
The first wide-character string is less than the second.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 11
Enter a value for x (in radians): 0.5
sin(0.50) = 0.479426
asin(sin(0.50)) = 0.500000
cos(0.50) = 0.877583
acos(cos(0.50)) = 0.500000
tan(0.50) = 0.546302
Verification: sin(asin(0.50)) = 0.500000
```

```
E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 12
Enter a number: 14578
The reversed integer is: 87541
E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 13
Enter a string: 12345x
The string contains non-digit characters.

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 14
Enter a string: Tanishq Ganesh Tote
Reversed string: etoT hsenaG qhsinaT

E:\COEP\DSA\Assignments\StringLabAssignment>a
Enter Question Number (1-15): 15
Enter a string: Tanishq Ganesh Tote
Enter a character: s
Result after cut: Tani
```