CSA02- C Programming Assignment 3

C. MITHILNANDHAN

192224037

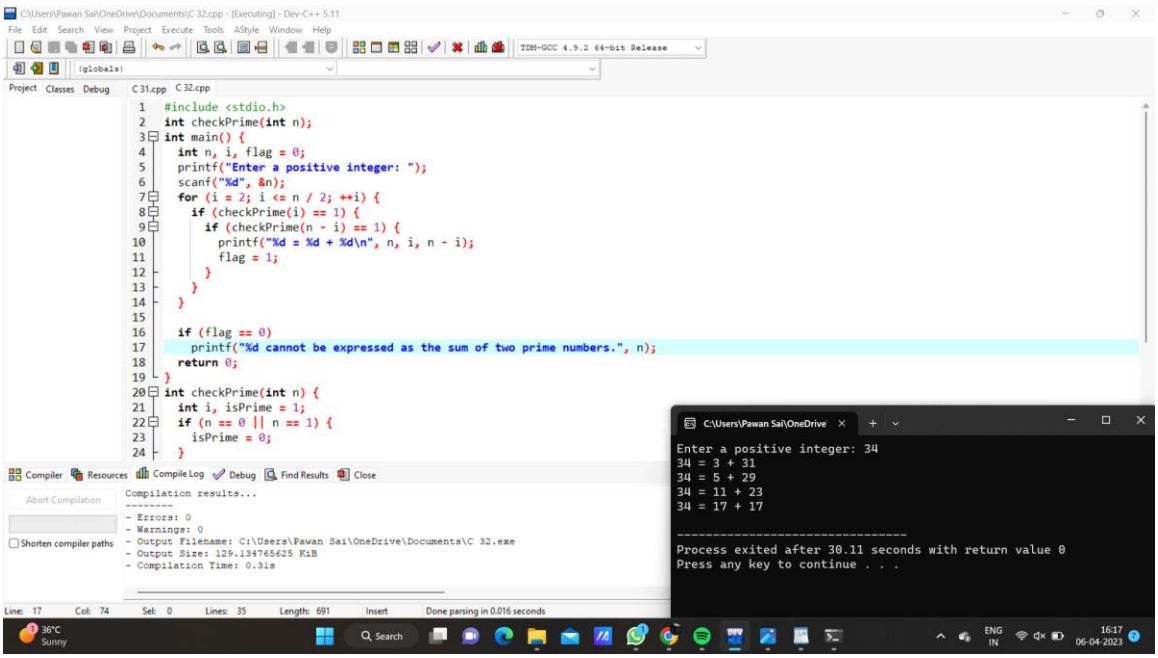
1. C Program to Display Prime Numbers Between Intervals Using Function

```c
#include <stdio.h>
int checkPrimeNumber(int n);
int main() {

  int n1, n2, i, flag;

  printf("Enter two positive integers: ");
  scanf("%d %d", &n1, &n2);
  if (n1 > n2) {
    n1 = n1 + n2;
    n2 = n1 - n2;
    n1 = n1 - n2;
  }

  printf("Prime numbers between %d and %d are: ", n1, n2);
  for (i = n1 + 1; i < n2; ++i) {
    flag = checkPrimeNumber(i);

    if (flag == 1) {
      printf("%d ", i);
    }
  }

  return 0;
}
int checkPrimeNumber(int n) {
  int j, flag = 1;

  for (j = 2; j <= n / 2; ++j) {

    if (n % j == 0) {
      flag = 0;
      break;
    }
  }

  return flag;
}
```

output :

```
Enter two positive integers: 12,30
Prime numbers between 12 and 47 are: 13 17 19 23 29 31 37 41 43
--------------------------------
Process exited after 5.696 seconds with return value 0
Press any key to continue . . .
```

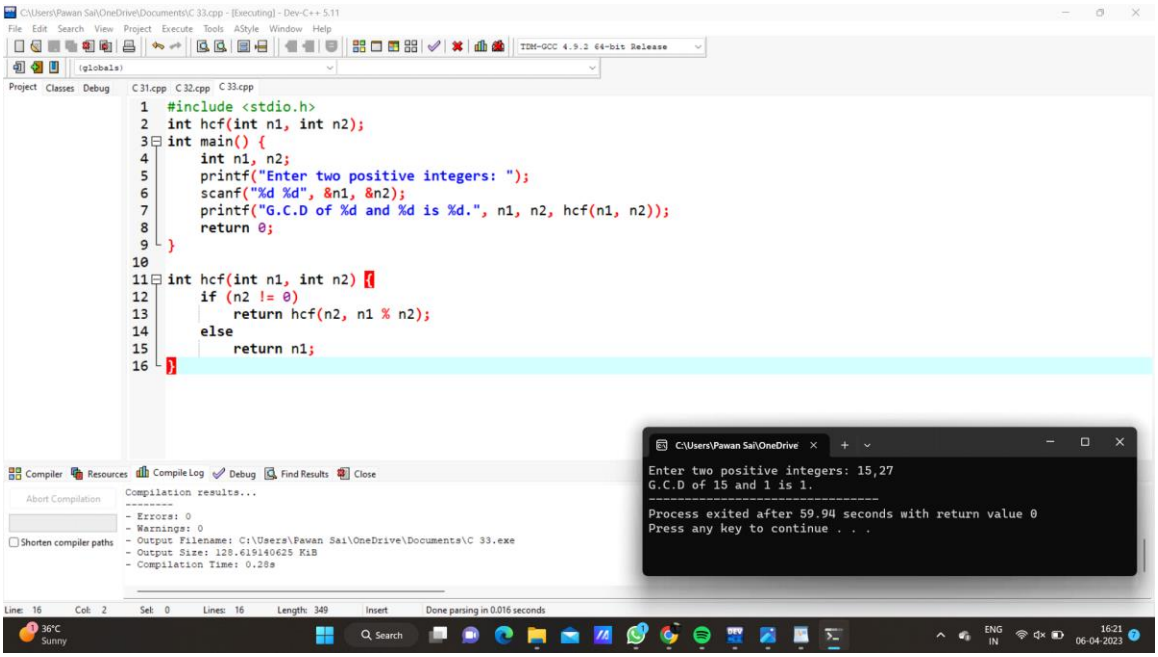
2. C Program to Check Whether a Number can be Expressed as Sum of Two Prime Numbers



```c
#include <stdio.h>
int checkPrime(int n);
int main() {
  int n, i, flag = 0;
  printf("Enter a positive integer: ");
  scanf("%d", &n);
  for (i = 2; i <= n / 2; ++i) {
    if (checkPrime(i) == 1) {
      if (checkPrime(n - i) == 1) {
        printf("%d = %d + %d\n", n, i, n - i);
        flag = 1;
      }
    }
  }

  if (flag == 0)
    printf("%d cannot be expressed as the sum of two prime numbers.", n);
  return 0;
}
int checkPrime(int n) {
  int i, isPrime = 1;
  if (n == 0 || n == 1) {
    isPrime = 0;
  }
```

```
Enter a positive integer: 34
34 = 3 + 31
34 = 5 + 29
34 = 11 + 23
34 = 17 + 17

Process exited after 30.11 seconds with return value 0
Press any key to continue . . .
```

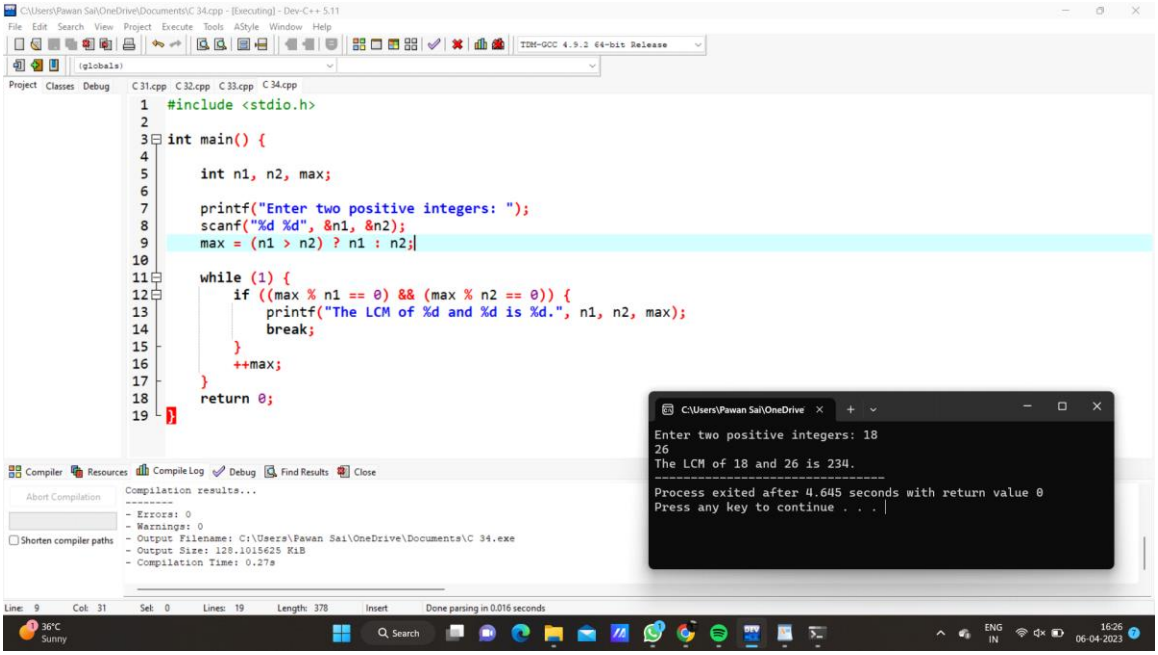
3. C Program to Find GCD of Two Numbers using Recursion



```c
#include <stdio.h>
int hcf(int n1, int n2);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);
    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
    return 0;
}

int hcf(int n1, int n2) {
    if (n2 != 0)
        return hcf(n2, n1 % n2);
    else
        return n1;
}
```

```
Enter two positive integers: 15,27
G.C.D of 15 and 1 is 1.

Process exited after 59.94 seconds with return value 0
Press any key to continue . . .
```

4. C Program to Find LCM of Two Numbers



5. C Program to Find Highest Frequency Character in a String

```c
#include <stdio.h>
#include <string.h>
char string1[100], visited[100];
int count[100] = {0}, flag = 0;
int main()
{
    int i, j = 0, k = 0, l, max, index;
    printf("Enter a string : ");
    scanf("%[^\n]s", string1);
    l = strlen(string1);
    for (i = 0; i < l; i++)
    {
        if (i == 0)
        {
            visited[j++] = string1[i];
            count[j - 1]++;
        }
        else
        {
            for (k = 0; k  < j; k++)
            {
                if (string1[i] == visited[k])
                {
                    count[k]++;
                    flag = 1;
                }
            }
            if (flag == 0)
            {
                visited[j++] = string1[i];
                count[j - 1]++;
            }
            flag = 0;
        }
    }

    for (i = 0; i < j; i++)
    {
        if ((i == 0) && (visited[i] != ' '))
        {
            max = count[i];
            continue;
        }
        if ((max < count[i]) && (visited[i] != ' '))
        {
            max = count[i];
            index = i;
        }
    }

    printf("\nMax repeated character in the string = %c ", visited[index]);
    printf("\nIt occurs %d times", count[index]);
}
```

OUTPUT:

```
Enter a string : Welcome to sanfoundry's C programming clas
s !

Max repeated character in the string = o
It occurs 4 times
--------------------------------
Process exited after 49.55 seconds with return value 0
Press any key to continue . . .
```

6. Anagram Program in C: Two strings are said to be anagrams if they satisfy two conditions, the length of both strings must be equal to each other and second the strings must have the same set of characters.

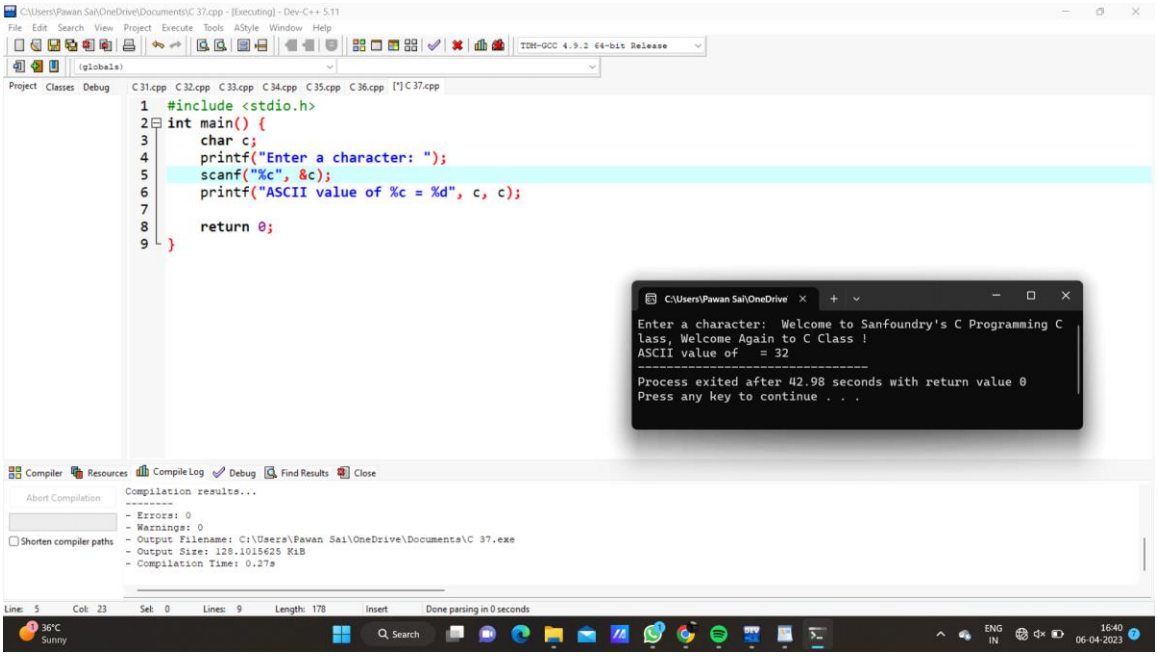
```c
#include <stdio.h>
 int get_anagrm (char [],char []);


int main ()
{
    char arr1 [50], arr2 [50];
    int count;

    printf (" Enter the first string: \n ");
    scanf (" %s", arr1);


    printf (" Enter the second string: \n ");
    scanf (" %s", arr2);
    count = get_anagrm (arr1, arr2);
    if (count == 1)
    {
        printf (" %s and %s strings are an anagram of each other. \n", arr1, arr2);
    }
    else
```

```c
    {
        printf (" %s and %s strings are not an anagram of each other. \n", arr1, arr2);
    }

    return 0;
}

int get_anagrm (char arr1[], char arr2[])
{
    int num1[20] = {0}, num2[20] = {0}, i = 0;
    while (arr1[i] != '\0')
    {
        num1[arr1[i] -'a']++;
        i++;
    }

    i = 0;
    while (arr2[i] != '\0')
    {
        num2[arr2[i] -'a']++;
```

```c
        i++;
    }

    for ( i = 0;i <20; i++)
    {
        if ( num1[i] != num2[i])
            return 0;
    }
    return 1;
}
```

```
 Enter the first string:
 hectare
 Enter the second string:
 teacher
 hectare and teacher strings are an anagram of each other.

--------------------------------
Process exited after 133.4 seconds with return value 0
Press any key to continue . . .
```

7. C Program to Find the Sum of ASCII Value of All Characters in the String



8. C Program to Print All Permutations of a Given String

SAN

```c
#include <stdio.h>
#include <string.h>
void swap(char* x, char* y)
{
    char temp;
    temp = *x;
    *x = *y;
    *y = temp;
}
void permute(char* a, int l, int r)
{
    int i;
    if (l == r)
        printf("%s\n", a);
    else
    {
        for (i = l; i <= r; i++)
        {
            swap((a + l), (a + i));
            permute(a, l + 1, r);
            swap((a + l), (a + i));

        }
    }
}
int main()
{
    char str[] = "SAN";
    int n = strlen(str);
    permute(str, 0, n - 1);
    return 0;
}
```

OUTPUT:

```
SAN
SNA
ASN
ANS
NAS
NSA

--------------------------------
Process exited after 2.466 seconds with return value 0
Press any key to continue . . .
```

9. Write a C program to copy one array elements to another array using pointers.

How to copy array elements from one array to another array using pointers. Logic to copy one array to another array using pointers in C programming.

```c
#include <stdio.h>
#define MAX_SIZE 100
void printArray(int arr[], int size);
int main()
{
    int source_arr[MAX_SIZE], dest_arr[MAX_SIZE];
    int size, i;
    int *source_ptr = source_arr;
    int *dest_ptr   = dest_arr;
    int *end_ptr;
    printf("Enter size of array: ");
    scanf("%d", &size);
    printf("Enter elements in array: ");
    for (i = 0; i < size; i++)
    {
        scanf("%d", (source_ptr + i));
    }
    end_ptr = &source_arr[size - 1];
    printf("\nSource array before copying: ");
    printArray(source_arr, size);
    printf("\nDestination array before copying: ");
    printArray(dest_arr, size);
    while(source_ptr <= end_ptr)
    {
        *dest_ptr = *source_ptr;
        source_ptr++;
        dest_ptr++;
    }
    printf("\n\nSource array after copying: ");
    printArray(source_arr, size);
    printf("\nDestination array after copying: ");
    printArray(dest_arr, size);
    return 0;
}
void printArray(int *arr, int size)
{
    int i;
    for (i = 0; i < size; i++)
    {
        printf("%d, ", *(arr + i));
    }
}
```

OUTPUT:

```
Enter size of array: 10
Enter elements in array:  10 -1 100 90 87 0 15 10 20 30

Source array before copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 30,
Destination array before copying: 6881394, 6619254, 1845494127, 6488175, 7143541, 7209061, 1509949530, 4391004, 3342368,
 3014713,

Source array after copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 30,
Destination array after copying: 10, -1, 100, 90, 87, 0, 15, 10, 20, 30,
--------------------------------
Process exited after 12.57 seconds with return value 0
Press any key to continue . . .
```

10. Write a C Program to reverse string using pointers and function.