

```

1  #include <stdio.h>
2
3  // Function to check whether a number is prime or not
4  int isPrime(int num) {
5      int i;
6      for(i = 2; i <= num/2; i++) {
7          if(num % i == 0) {
8              return 0; // Not a prime number
9          }
10     }
11     return 1; // Prime number
12 }
13
14 int main() {
15     int start, end, i;
16
17     printf("Enter two positive integers: ");
18     scanf("%d %d", &start, &end);
19
20     printf("Prime numbers between %d and %d are: ", start, end);
21
22     // Loop through all the numbers between the start and end intervals
23     for(i = start; i <= end; i++) {
24         if(isPrime(i)) {
25             printf("%d ", i);
26         }
27     }
28
29     return 0;
30 }

```

```

C:\Users\91961\Documents\1.assing 3.cpp
Enter two positive integers: 12 30
Prime numbers between 12 and 30 are: 13 17 19 23 29
-----
Process exited after 14.04 seconds with return value 0
Press any key to continue . . .

```

2.assi 3.cpp

```
3 int checkPrime(int n); // function to check whether a number is prime
4
5 int main() {
6     int n, i, flag = 0;
7     printf("Enter a positive integer: ");
8     scanf("%d", &n);
9     for (i = 2; i <= n/2; ++i) {
10        // condition for i to be prime
11        if (checkPrime(i) == 1) {
12            // condition for n-i to be prime
13            if (checkPrime(n-i) == 1) {
14                printf("%d = %d + %d\n", n, i, n-i);
15                flag = 1;
16            }
17        }
18    }
19    if (flag == 0)
20        printf("%d cannot be expressed as the sum of two prime numbers.", n);
21    return 0;
22 }
23
24 // function to check whether a number is prime
25 int checkPrime(int n) {
26     int i, isPrime = 1;
27     for (i = 2; i <= n/2; ++i) {
28         if (n % i == 0) {
29             isPrime = 0;
30             break;
31         }
32     }
33     return isPrime;
34 }
```

```
C:\Users\91961\Documents\2. x + v - □ ×
Enter a positive integer: 34
34 = 3 + 31
34 = 5 + 29
34 = 11 + 23
34 = 17 + 17

-----
Process exited after 8.989 seconds with return value 0
Press any key to continue . . . |
```

```

1  #include <stdio.h>
2
3  int gcd(int a, int b);
4
5  int main()
6  {
7      int num1, num2, result;
8      printf("Enter two numbers: ");
9      scanf("%d %d", &num1, &num2);
10     result = gcd(num1, num2);
11     printf("GCD of %d and %d is %d\n", num1, num2, result);
12     return 0;
13 }
14
15 int gcd(int a, int b)
16 {
17     if (b == 0)
18         return a;
19     else
20         return gcd(b, a % b);
21 }

```

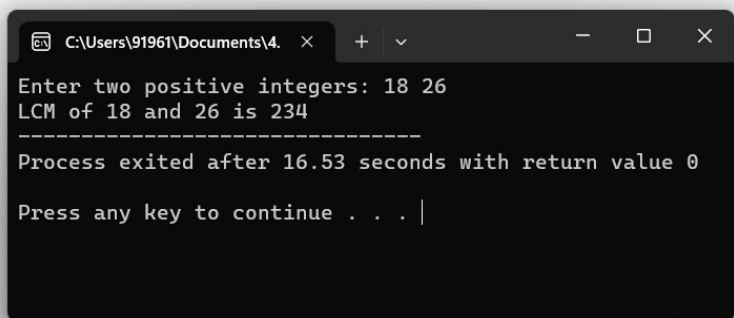
```

C:\Users\91961\Documents\3.  x  +  v  -  □  ×
Enter two numbers: 3 78
GCD of 3 and 78 is 3

-----
Process exited after 12.39 seconds with return value 0
Press any key to continue . . . |

```

```
4.assi3.cpp
1  #include <stdio.h>
2
3  int main()
4  {
5      int num1, num2, max;
6      printf("Enter two positive integers: ");
7      scanf("%d %d", &num1, &num2);
8      max = (num1 > num2) ? num1 : num2;
9      while(1)
10     {
11         if(max % num1 == 0 && max % num2 == 0)
12         {
13             printf("LCM of %d and %d is %d", num1, num2, max);
14             break;
15         }
16         ++max;
17     }
18     return 0;
19 }
```

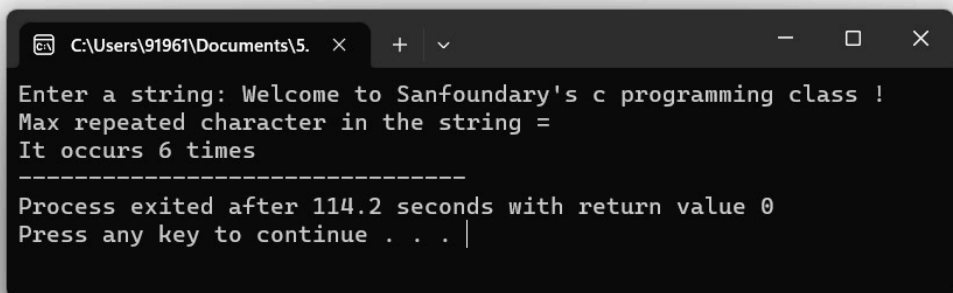


A screenshot of a terminal window with a dark background. The window title bar shows the file path 'C:\Users\91961\Documents\4.' and standard window controls. The terminal output shows the program's execution: it prompts for two positive integers, receives '18 26', calculates the LCM as 234, and prints the result. It then displays a separator line, a message about the process exiting after 16.53 seconds, and a prompt to press any key to continue.

```
C:\Users\91961\Documents\4.
Enter two positive integers: 18 26
LCM of 18 and 26 is 234
-----
Process exited after 16.53 seconds with return value 0
Press any key to continue . . .
```

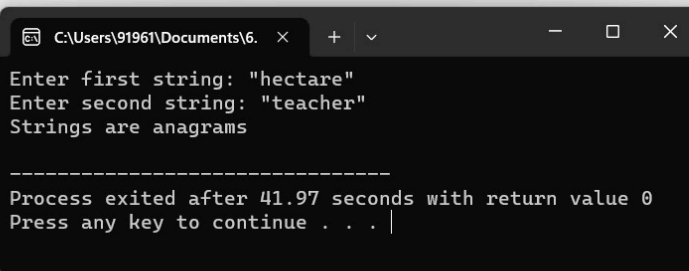
5.assi 3.cpp

```
1 #include <stdio.h>
2 #include <string.h>
3
4 #define MAX_SIZE 100
5
6 int main()
7 {
8     char str[MAX_SIZE];
9     int i, j, freq[MAX_SIZE];
10    int maxFreq, maxIndex;
11
12    printf("Enter a string: ");
13    fgets(str, MAX_SIZE, stdin);
14
15    int len = strlen(str);
16    if (str[len - 1] == '\n')
17        str[len - 1] = '\0';
18
19    // initialize frequency array
20    for (i = 0; i < strlen(str); i++)
21        freq[i] = 0;
22
23    // count frequency of each character
24    for (i = 0; i < strlen(str); i++) {
25        for (j = 0; j < strlen(str); j++) {
26            if (str[i] == str[j])
27                freq[i]++;
28        }
29    }
30
31    // find character with highest frequency
32    maxFreq = -1;
33    for (i = 0; i < strlen(str); i++) {
34        if (freq[i] > maxFreq) {
35            maxFreq = freq[i];
36            maxIndex = i;
37        }
38    }
39
40    printf("Max repeated character in the string = %c\n", str[maxIndex]);
41    printf("It occurs %d times", maxFreq);
42
43    return 0;
44 }
```



6. assi 3.cpp

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int main()
5 {
6     char str1[100], str2[100];
7     int len1, len2, i, j, freq[256];
8
9     // Get first string input
10    printf("Enter first string: ");
11    fgets(str1, 100, stdin);
12    len1 = strlen(str1);
13
14    // Get second string input
15    printf("Enter second string: ");
16    fgets(str2, 100, stdin);
17    len2 = strlen(str2);
18
19    // Check if both strings have same length
20    if (len1 != len2) {
21        printf("Strings are not anagrams\n");
22        return 0;
23    }
24
25    // Initialize frequency array with all zeros
26    memset(freq, 0, sizeof(freq));
27
28    // Count frequency of characters in first string
29    for (i = 0; i < len1; i++) {
30        freq[(int)str1[i]]++;
31    }
32
33    // Subtract frequency of characters in second string
34    for (j = 0; j < len2; j++) {
35        freq[(int)str2[j]]--;
36    }
37
38    // Check if all frequencies are zero
39    for (i = 0; i < 256; i++) {
40        if (freq[i] != 0) {
41            printf("Strings are not anagrams\n");
42            return 0;
43        }
44    }
45
46    printf("Strings are anagrams\n");
47
48    return 0;
49 }
```



```
C:\Users\91961\Documents\6. × + ▾
Enter first string: "hectare"
Enter second string: "teacher"
Strings are anagrams

-----
Process exited after 41.97 seconds with return value 0
Press any key to continue . . . |
```

```

7.assi 3.cpp
1 #include <stdio.h>
2 #include <string.h>
3
4 int main()
5 {
6     char str[100];
7     int sum = 0, i;
8
9     printf("Enter a string: ");
10    gets(str);
11
12    for(i = 0; i < strlen(str); i++)
13    {
14        sum += (int)str[i]; // add the ASCII value of each character to the sum
15    }
16
17    printf("Sum of ASCII values of all characters in the string: %d", sum);
18
19    return 0;
20 }

```

```

C:\Users\91961\Documents\7.
Enter a string: Welcome to Sanfoundary's C Programming Class,Welcome Again to C Class !
Sum of ASCII values of all characters in the string: 6372
-----
Process exited after 147.9 seconds with return value 0
Press any key to continue . . .

```

8.assin 3.cpp

```
1  #include <stdio.h>
2  #include <string.h>
3
4  /* Function to swap two characters */
5  void swap(char *x, char *y)
6  {
7      char temp;
8      temp = *x;
9      *x = *y;
10     *y = temp;
11 }
12
13 /* Function to print permutations of a given string */
14 void permute(char *str, int start, int end)
15 {
16     if (start == end) {
17         printf("%s\n", str);
18     } else {
19         int i;
20         for (i = start; i <= end; i++) {
21             swap((str+start), (str+i));
22             permute(str, start+1, end);
23             swap((str+start), (str+i)); // backtrack
24         }
25     }
26 }
27
28 /* Main function */
29 int main()
30 {
31     char str[100];
32     printf("Enter a string: ");
33     scanf("%s", str);
34     int n = strlen(str);
35     printf("All permutations of %s are:\n", str);
36     permute(str, 0, n-1);
37     return 0;
38 }
```

```
C:\Users\91961\Documents\8. x + v - □ ×
Enter a string: SAM
All permutations of SAM are:
SAM
SMA
ASM
AMS
MAS
MSA

-----
Process exited after 22.48 seconds with return value 0
Press any key to continue . . . |
```


9.ASSI 3.cpp

```
1 #include <stdio.h>
2
3 #define SIZE 5
4
5 int main() {
6     int arr1[SIZE] = {1, 2, 3, 4, 5};
7     int arr2[SIZE];
8     int *ptr1 = arr1; // pointer to arr1
9     int *ptr2 = arr2; // pointer to arr2
10
11     // copy elements from arr1 to arr2 using pointers
12     for (int i = 0; i < SIZE; i++) {
13         *(ptr2 + i) = *(ptr1 + i);
14     }
15
16     // print arr2 to confirm elements were copied successfully
17     for (int i = 0; i < SIZE; i++) {
18         printf("%d ", arr2[i]);
19     }
20
21     return 0;
22 }
```

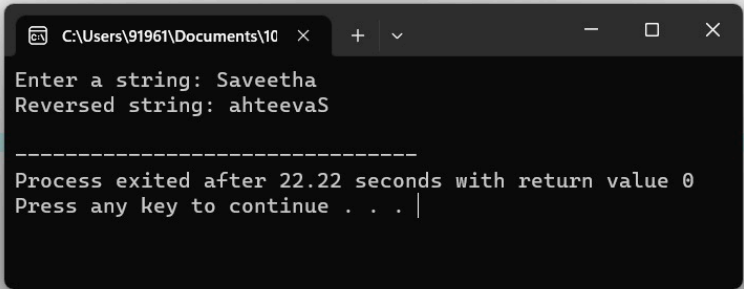
C:\Users\91961\Documents\9.

1 2 3 4 5

Process exited after 3.91 seconds with return value 0
Press any key to continue . . .

10.Assi 3.cpp

```
1  #include <stdio.h>
2  #include <string.h>
3
4  void reverse(char *);
5
6  int main() {
7      char str[100];
8
9      printf("Enter a string: ");
10     scanf("%s", str);
11
12     reverse(str);
13
14     printf("Reversed string: %s\n", str);
15
16     return 0;
17 }
18
19 void reverse(char *str) {
20     int i, j;
21     char temp;
22
23     i = 0;
24     j = strlen(str) - 1;
25
26     while (i < j) {
27         temp = *(str + i);
28         *(str + i) = *(str + j);
29         *(str + j) = temp;
30         i++;
31         j--;
32     }
33 }
```



```
C:\Users\91961\Documents\10
Enter a string: Saveeetha
Reversed string: ahteevaS

-----
Process exited after 22.22 seconds with return value 0
Press any key to continue . . .
```