Reg. No: 21BCE1297 **Name: Vidhi Shah** **Date: 10/03/22**

PPS6

Q1

Aim:

Write a ‘C’ Program to check whether the i/p number is palindrome or not?

Procedure:

Input:
A number, n

Output:
Palindrome or not a palindrome

Algorithm:

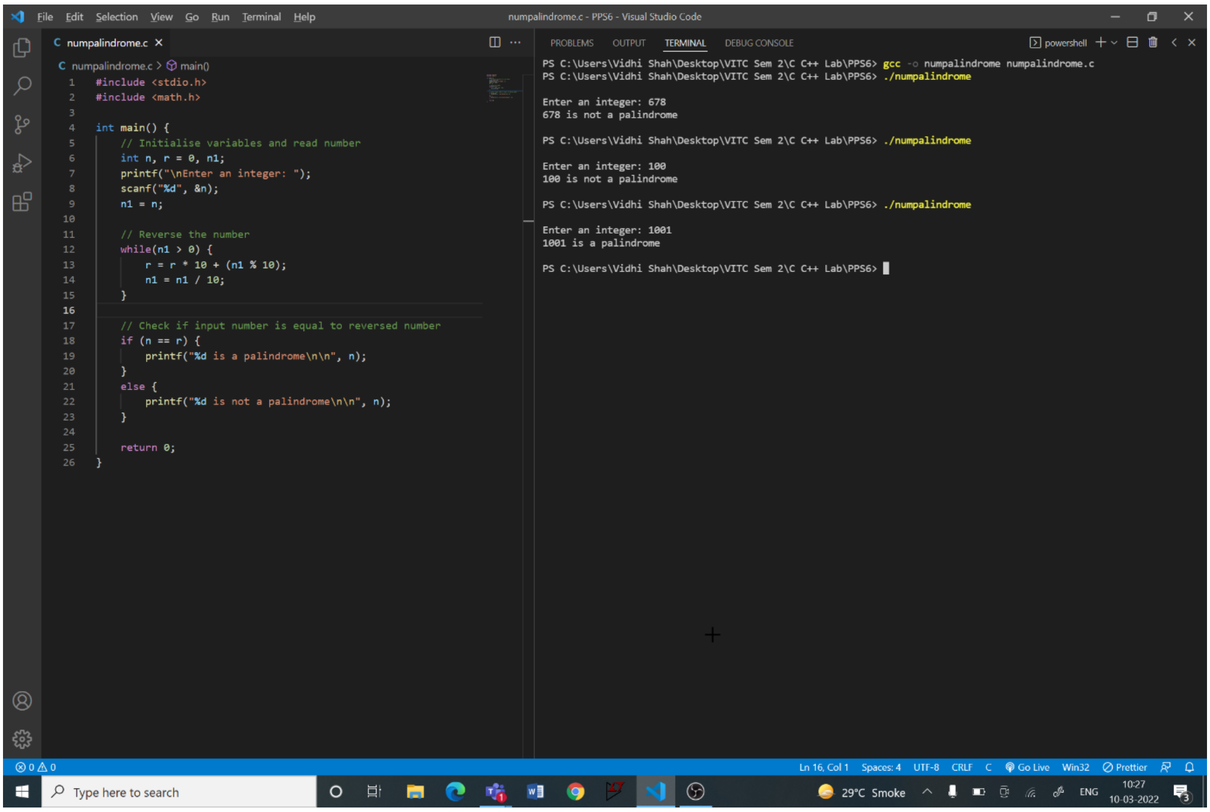
Method 1

```
Step 1: Initialise variables and read n as an integer
Step 2: Let reverse number, $r = 0$ and $n1 = n$
Step 3: While $n1 > 0$
            $r = r*10 + (n1\%10)$
            Integer division of n1 by 10
Step 4: If the reverse, r is equal to n
            Print n is a palindrome
Step 5: If the reverse, r is not equal to n
            Print n is not a palindrome
```

Method 2

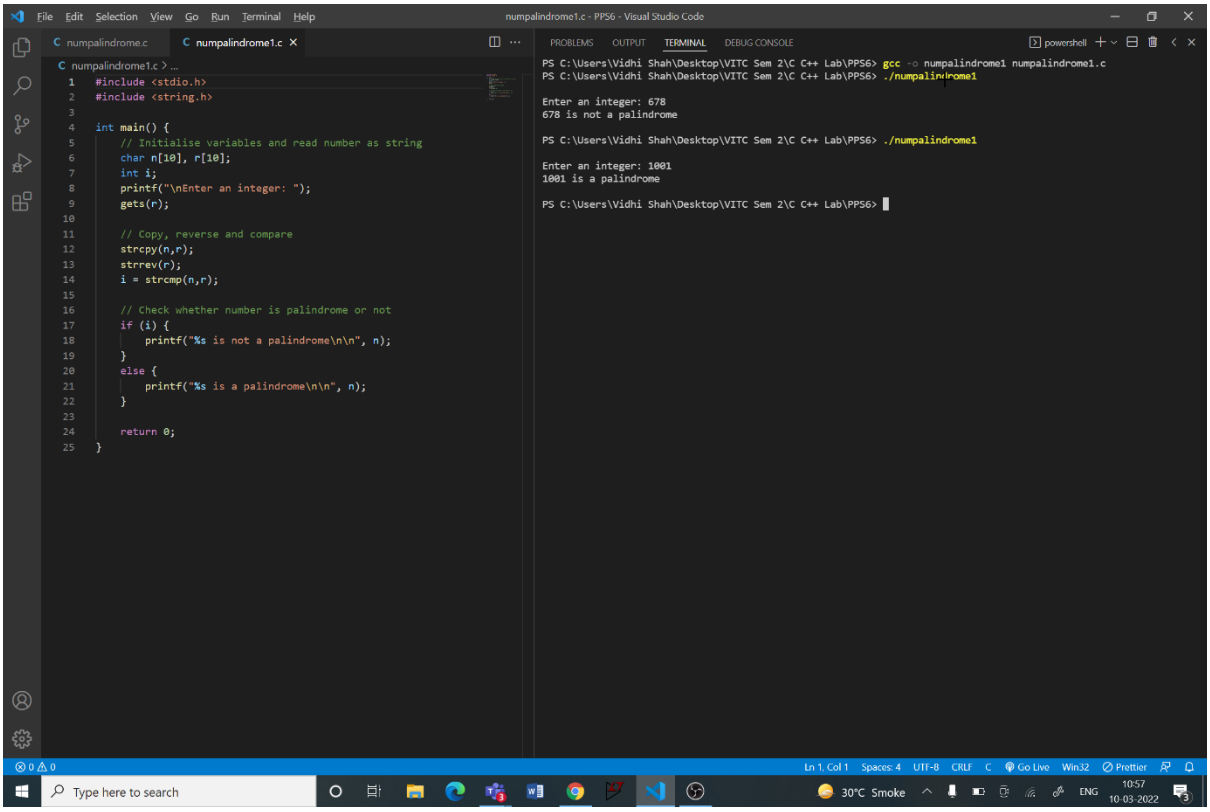
```
Step 1: Initialise variables and read n as a string
Step 2: Copy the string from n to r (reverse string)
Step 3: Reverse the string r
Step 4: Compare the 2 strings n and r and store the integer result in a variable, i
Step 5: If i is equal 0
            Print n is a palindrome
Step 6: If i is not equal to 0
            Print n is not a palindrome
```

Code:


File Edit Selection View Go Run Terminal Help
numpalindrome.c - PPS6 - Visual Studio Code
numpalindrome.c
numpalindrome.c > main()
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
powershell
Ln 16, Col 1 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
29°C Smoke ENG 10:27 10-03-2022

```c
#include <stdio.h>
#include <math.h>

int main() {
    // Initialise variables and read number
    int n, r = 0, n1;
    printf("\nEnter an integer: ");
    scanf("%d", &n);
    n1 = n;

    // Reverse the number
    while(n1 > 0) {
        r = r * 10 + (n1 % 10);
        n1 = n1 / 10;
    }

    // Check if input number is equal to reversed number
    if (n == r) {
        printf("%d is a palindrome\n\n", n);
    }
    else {
        printf("%d is not a palindrome\n\n", n);
    }

    return 0;
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> gcc -o numpalindrome numpalindrome.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./numpalindrome

Enter an integer: 678
678 is not a palindrome

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./numpalindrome

Enter an integer: 100
100 is not a palindrome

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./numpalindrome

Enter an integer: 1001
1001 is a palindrome

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> 
```


File Edit Selection View Go Run Terminal Help
numpalindrome1.c - PPS6 - Visual Studio Code
numpalindrome.c
numpalindrome1.c
numpalindrome1.c > ...
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
powershell
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
30°C Smoke ENG 10:57 10-03-2022

```c
#include <stdio.h>
#include <string.h>

int main() {
    // Initialise variables and read number as string
    char n[10], r[10];
    int i;
    printf("\nEnter an integer: ");
    gets(r);

    // Copy, reverse and compare
    strcpy(n,r);
    strrev(r);
    i = strcmp(n,r);

    // Check whether number is palindrome or not
    if (i) {
        printf("%s is not a palindrome\n\n", n);
    }
    else {
        printf("%s is a palindrome\n\n", n);
    }

    return 0;
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> gcc -o numpalindrome1 numpalindrome1.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./numpalindrome1

Enter an integer: 678
678 is not a palindrome

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./numpalindrome1

Enter an integer: 1001
1001 is a palindrome

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> 
```

Q2

Aim:

Write a program to display the following pattern:

1
2 2
3 3 3
4 4 4 4

Procedure:

Input:

A number, n

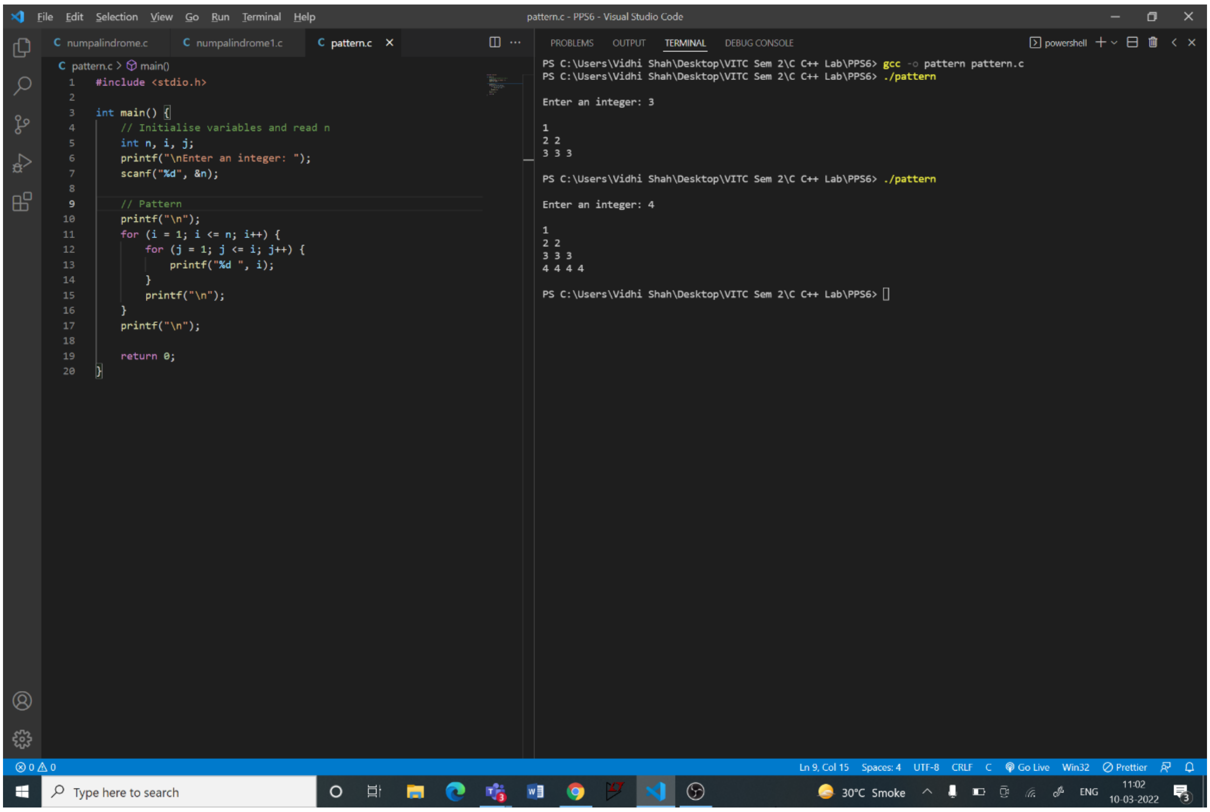
Output:

Pattern

Algorithm:

```
Step 1: Initialise variables and read n
Step 2: Initialise i to 1. Until i is less than or equal to n
            Initialise j to 1. Until j is less than or equal to i
                Print i
                Increment j by 1
            Go to next line
            Increment i by 1
```

Code:



```c
#include <stdio.h>

int main() {
    // Initialise variables and read n
    int n, i, j;
    printf("\nEnter an integer: ");
    scanf("%d", &n);

    // Pattern
    printf("\n");
    for (i = 1; i <= n; i++) {
        for (j = 1; j <= i; j++) {
            printf("%d ", i);
        }
        printf("\n");
    }
    printf("\n");

    return 0;
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> gcc -o pattern pattern.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./pattern

Enter an integer: 3

1
2 2
3 3 3

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./pattern

Enter an integer: 4

1
2 2
3 3 3
4 4 4 4

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> 
```

Q3

Aim:

Given n number of elements containing positive, negative, Zero’s, odd and even numbers, write a C program to print the number of positive, negative, Zero’s, odd and even numbers in the array.

Procedure:

Input:

Number of elements, n
Next n lines contain n numbers

Output:

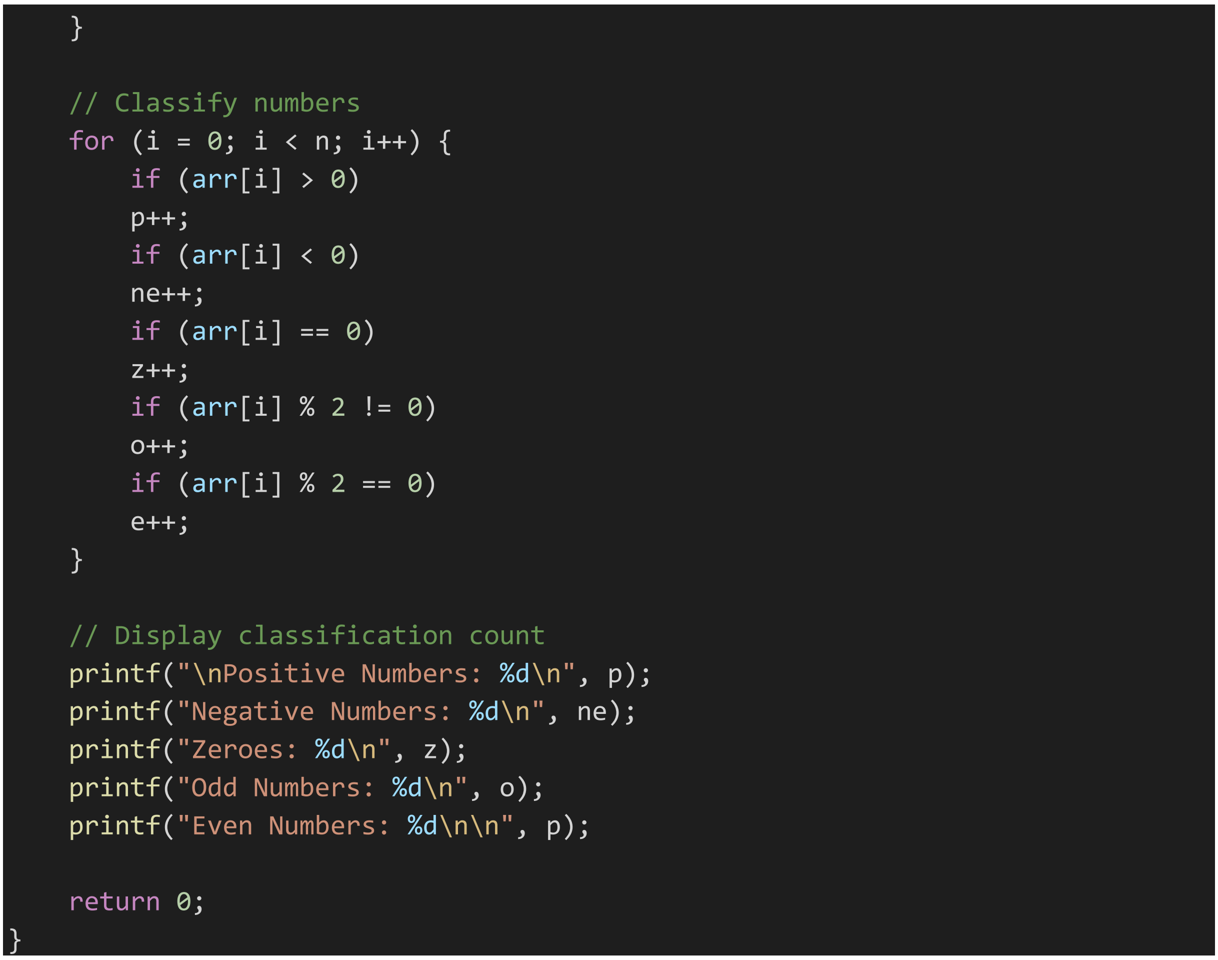
Number of positive, negative, zero, odd and even elements

Algorithm:

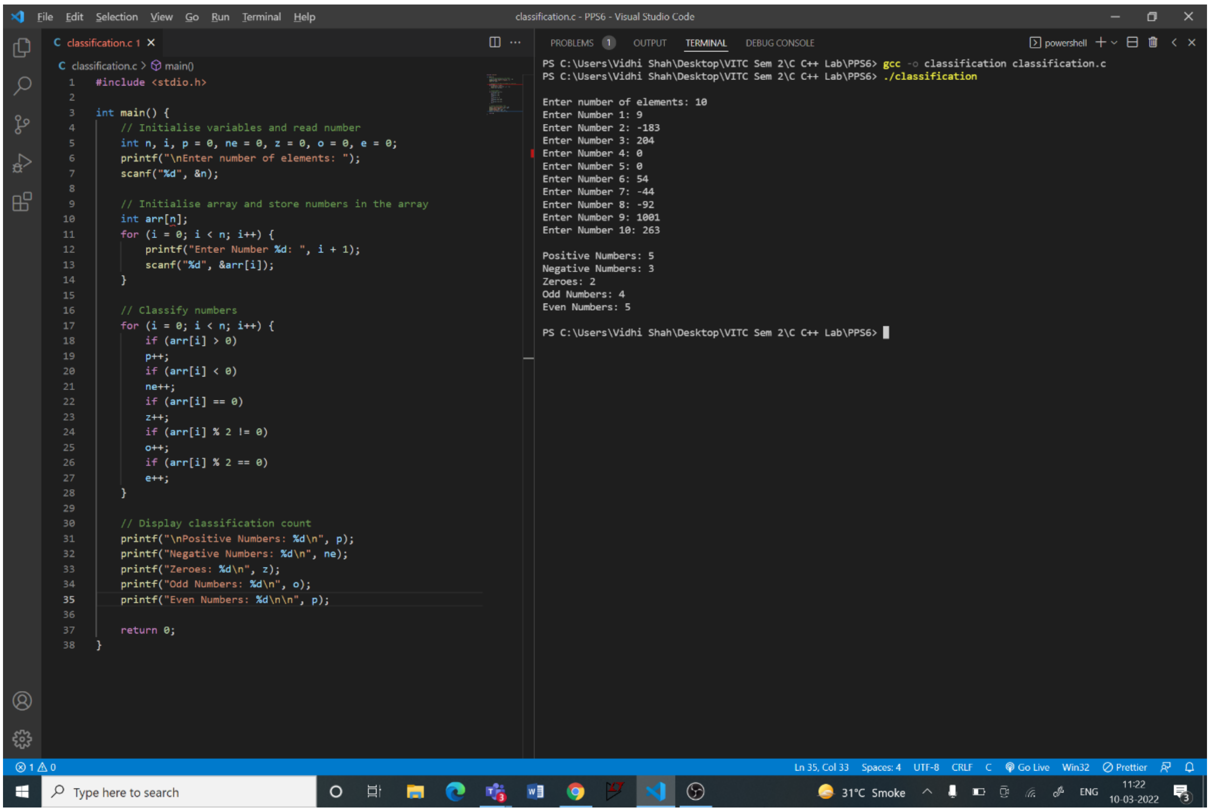
```
Step 1: Initialise variables and read n
Step 2: Initialise array of size n
Step 3: Using for loop read each number and store it in the array
Step 4: For each number in the array
            If number is greater than 0
                Increment positive count by 1
            If number is less than 0
                Increment negative count by 1
            If number is equal to 0
                Increment zero count by 1
            If number is not divisible by 2
                Increment odd count by 1
            If number is divisible by 2
                Increment even count by 1
Step 5: Display all the counts in separate lines
```

Code:

```
#include <stdio.h>

int main() {
    // Initialise variables and read number
    int n, i, p = 0, ne = 0, z = 0, o = 0, e = 0;
    printf("\nEnter number of elements: ");
    scanf("%d", &n);

    // Initialise array and store numbers in the array
    int arr[n];
    for (i = 0; i < n; i++) {
        printf("Enter Number %d: ", i + 1);
        scanf("%d", &arr[i]);
```



```c
    }

    // Classify numbers
    for (i = 0; i < n; i++) {
        if (arr[i] > 0)
        p++;
        if (arr[i] < 0)
        ne++;
        if (arr[i] == 0)
        z++;
        if (arr[i] % 2 != 0)
        o++;
        if (arr[i] % 2 == 0)
        e++;
    }

    // Display classification count
    printf("\nPositive Numbers: %d\n", p);
    printf("Negative Numbers: %d\n", ne);
    printf("Zeroes: %d\n", z);
    printf("Odd Numbers: %d\n", o);
    printf("Even Numbers: %d\n\n", p);

    return 0;
}
```



```c
#include <stdio.h>

int main() {
    // Initialise variables and read number
    int n, i, p = 0, ne = 0, z = 0, o = 0, e = 0;
    printf("\nEnter number of elements: ");
    scanf("%d", &n);

    // Initialise array and store numbers in the array
    int arr[n];
    for (i = 0; i < n; i++) {
        printf("Enter Number %d: ", i + 1);
        scanf("%d", &arr[i]);
    }

    // Classify numbers
    for (i = 0; i < n; i++) {
        if (arr[i] > 0)
        p++;
        if (arr[i] < 0)
        ne++;
        if (arr[i] == 0)
        z++;
        if (arr[i] % 2 != 0)
        o++;
        if (arr[i] % 2 == 0)
        e++;
    }

    // Display classification count
    printf("\nPositive Numbers: %d\n", p);
    printf("Negative Numbers: %d\n", ne);
    printf("Zeroes: %d\n", z);
    printf("Odd Numbers: %d\n", o);
    printf("Even Numbers: %d\n\n", p);

    return 0;
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> gcc -o classification classification.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./classification

Enter number of elements: 10
Enter Number 1: 9
Enter Number 2: -183
Enter Number 3: 204
Enter Number 4: 0
Enter Number 5: 0
Enter Number 6: 54
Enter Number 7: -44
Enter Number 8: -92
Enter Number 9: 1001
Enter Number 10: 263

Positive Numbers: 5
Negative Numbers: 3
Zeroes: 2
Odd Numbers: 4
Even Numbers: 5

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6>
```

Q4

Aim:

In an orphanage there were about 5 to 7 babies in the age group of 2months to 1 year. The founder of the orphanage wanted to name all these babies with special names. The names were special since all the names to be selected are palindromes. The founder's friend gave him a list of baby names. The founder needs to select the palindrome names out of the list. Help the founder by implementing a C code in identifying the palindrome names out of the list. Also your program needs to count the number of palindrome names present in the list. If there is no palindrome string print 0.

Eg: If the list is

```
sachin
eve
arora
pilip
kamal
kumar
joshua
jones
nitin
emme
```

Your program has to print all palindrome strings.

```
eve
arora
pilip
nitin
emme
5 (number of palindrome strings present in the list)
```

Procedure:

Input:

Number of names, n
Next n lines contain n names

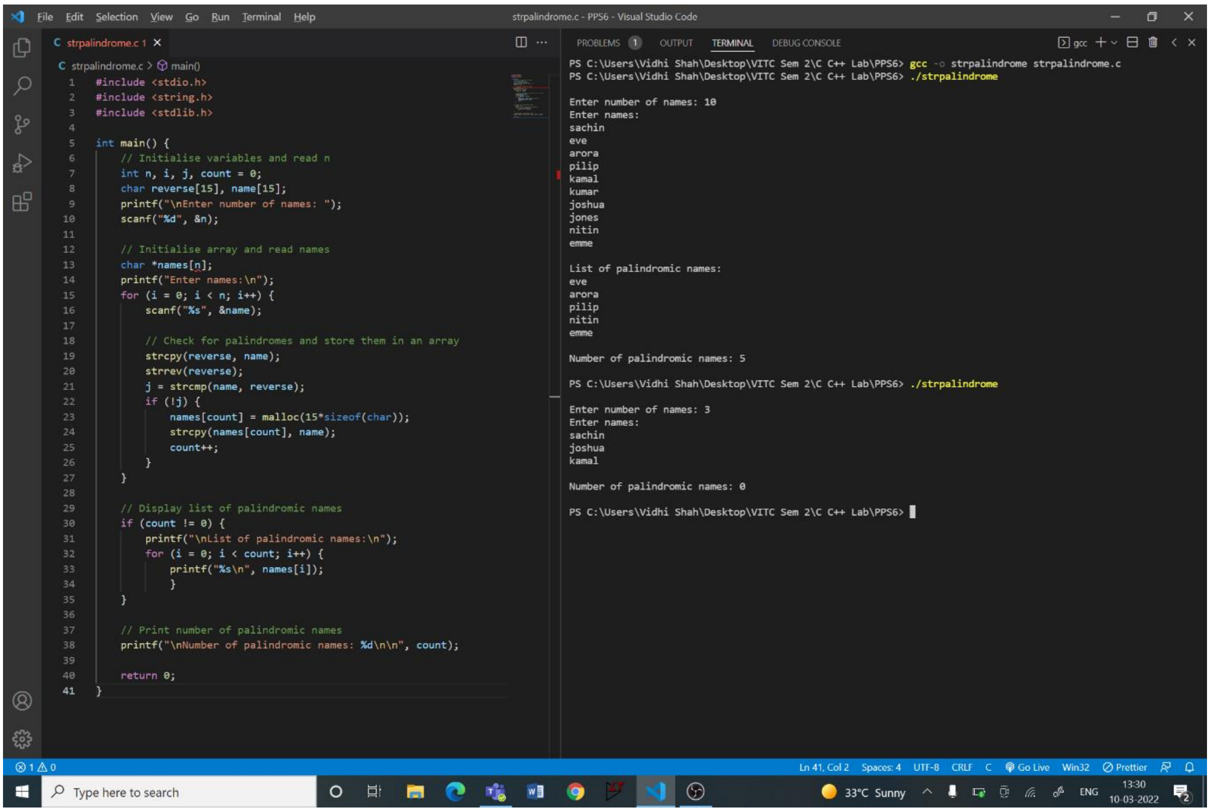
Output:

List of palindromic names
Number of palindromic names

Algorithm:

```
Step 1: Initialise variables and read n, initialise count to 0
Step 2: Initialise a character array (size n) of pointers (array of strings)
Step 3: Use for loop. For each iteration
            Read name
            Copy the string from name to reverse
            Reverse the string reverse
            Compare name and reverse and store the integer result in a variable, j
            If j is equal 0
                Allocate memory to count^th position of the array to store a string
                Copy the string name in count^th position of the array
                Increment count by 1
Step 6: Print all names stored in the array using for loop
Step 7: Print number of names printed
```

Code:



```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main() {
    // Initialise variables and read n
    int n, i, j, count = 0;
    char reverse[15], name[15];
    printf("\nEnter number of names: ");
    scanf("%d", &n);

    // Initialise array and read names
    char *names[n];
    printf("Enter names:\n");
    for (i = 0; i < n; i++) {
        scanf("%s", &name);

        // Check for palindromes and store them in an array
        strcpy(reverse, name);
        strrev(reverse);
        j = strcmp(name, reverse);
        if (!j) {
            names[count] = malloc(15*sizeof(char));
            strcpy(names[count], name);
            count++;
        }
    }

    // Display list of palindromic names
    if (count != 0) {
        printf("\nList of palindromic names:\n");
        for (i = 0; i < count; i++) {
            printf("%s\n", names[i]);
            }
    }

    // Print number of palindromic names
    printf("\nNumber of palindromic names: %d\n\n", count);

    return 0;
}
```

```
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> gcc -o strpalindrome strpalindrome.c
PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./strpalindrome

Enter number of names: 10
Enter names:
sachin
eve
arora
pilip
kamal
kumar
joshua
jones
nitin
emme

List of palindromic names:
eve
arora
pilip
nitin
emme

Number of palindromic names: 5

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> ./strpalindrome

Enter number of names: 3
Enter names:
sachin
joshua
kamal

Number of palindromic names: 0

PS C:\Users\Vidhi Shah\Desktop\VITC Sem 2\C C++ Lab\PPS6> 
```