1.Program to check whether given string is palindrome or not

```
#include <stdio.h>
#include <string.h>
#include <stdbool.h>
bool isPalindrome(char str[]) {
    int l = 0;
    int r = strlen(str) - 1;
    while (l < r) {
        if (str[l] != str[r])
        return false;
        l++;
        r--;

    }
    return true;
}
int main() {
    char str[100];
    printf("Enter a string: ");
    scanf("%s", str);
    if (isPalindrome(str))
    {
        printf("Palindrome\n");
    }
    else
    {
        printf("Not Palindrome\n");
    }
    return 0;

}
```

2.Program to reverse the string

```
#include<stdio.h>
#include<string.h>
void reverseString(char * str)
{
    int len = strlen(str);
    int i, j;
    char temp;

    for (i = 0; j = len - 1; i < j; i++; j--)
    {
        temp = str[i];
        str[i] = str[j];
        str[j] = temp;
    }
}
int main()
{
    char word[100];
    printf("Enter the word to be reversed");
    scanf("%s", word);
    reverseString(word);
    printf("Reversed word: ", word);
    return 0;
}
```

3. A customer support system locks multiple issue reported by the users the first unique word in the lock is often the most critical issue.Given a message find the first non repeating character which represents the first unique issue.If all the character repeats return -1.

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

#define MAX 200

int f_NonRep(const char *msg)
{
    int cnt[MAX] = {0};
    for (int i = 0; msg[i]; i++)
    {
        cnt[msg[i]]++;
    }

    for (int i = 0; msg[i]; i++)
    {
        if (cnt[msg[i]] == 1)
        {
            return msg[i];
        }
    }

    return -1;
}

int main()
{
    char input[MAX];

    printf("Enter the message: ");
    scanf("%255s", input);

    int result = f_NonRep(input);

    if (result != -1) \
    {
        printf("The first non-repeating character is: '%c'\n", result);
    }
    else
    {
        printf("-1\n");
    }

    return 0;
}
```

4.Sum of the number
Input:1234
Output:10

```
#include <stdio.h>

int main()
{
    char ch;
    int sum = 0;
    printf("Enter the string: ");
    while (scanf("%c", &ch) == 1 && ch != '\n')
    {
        if (ch >= '0' && ch <= '9') {
            sum += ch - '0';
        }
    }

    printf("The sum of the digits in the string is: %d\n", sum);

    return 0;
}
```

5. Using recursion
Input:1234
Output:10

```
#include <stdio.h>
int sum_ofDig(const char * str)
{
    if ( * str == '\0')
    {
        return 0;
    }
int sum = 0;
    if ( * str >= '0' && * str <= '9')
    {
        sum = * str - '0';
    }
    return sum + sum_ofDig(str + 1);
}

int main()
{
    char input[100];

    printf("Enter the string: ");
    scanf("%99s", input);
    int totalSum = sum_ofDig(input);
    printf("The sum of the digits in the string is: %d\n", totalSum);

    return 0;
}
```