```c
//Convert CurrencyS value into words

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


void TwoWords(char* num)
{
    int len = strlen(num);

    if (len == 0) {
        printf("empty string\n");
        return;
    }
    if (len > 4) {
        printf("Length more than 4 is not supported\n");
        return;
    }

    char* single_digits[]= { "zero", "one", "two",   "three", "four","five", "six", "seven", "eight", "nine" };

    char* two_digits[]= { "","ten","eleven","twelve","thirteen","fourteen","fifteen","sixteen","seventeen", "eighteen", "nineteen" };

    char* tens_multiple[] = { "","","twenty","thirty", "forty", "fifty","sixty","seventy", "eighty","ninety" };

    char* tens_power[] = { "hundred", "thousand" };

    printf("\n%s:", num);

    if (len == 1) {
        printf("%s\n", single_digits[*num - '0']);
        return;
    }

    while (*num != '\0') {

        if (len >= 3) {
            if (*num - '0' != 0) {
                printf("%s ", single_digits[*num - '0']);
                printf("%s ",
                       tens_power[len - 3]);
            }
            --len;
        }

        else {

            if (*num == '1') {
                int sum = *num - '0' + *(num + 1) - '0';
                printf("%s\n", two_digits[sum]);
                return;
            }

            else if (*num == '2' && *(num + 1) == '0') {
                printf("twenty\n");
                return;
            }

            else {
                int i = *num - '0';
                printf("%s ", i ? tens_multiple[i] : "");
                ++num;
                if (*num != '0')
                    printf("%s ",
                           single_digits[*num - '0']);
            }
        }
        ++num;
    }
}

int main(void)
{
    char str[20];
    scanf("%[^\n]%*c", str);
    TwoWords(str);
    return 0;
}
```

```
PS G:\C\Assignment4> cd "g:\C\Assignment4\" ; if ($?) { gcc Q1Currency_to_Word.c -o Q1Currency_to_Word } ; if ($?) { .\Q1Currency_to_Word }
350

350:three hundred fifty
PS G:\C\Assignment4>
```

```c
#include <stdio.h>
int main() {
  int n, reversed = 0, remainder, original;
    printf("Enter an integer: ");
    scanf("%d", &n);
    original = n;

    // reversed integer is stored in reversed variable
    while (n != 0) {
        remainder = n % 10;
        reversed = reversed * 10 + remainder;
        n /= 10;
    }

    // palindrome if orignal and reversed are equal
    if (original == reversed)
        printf("%d is a palindrome.", original);
    else
        printf("%d is not a palindrome.", original);

    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS G:\C\Assignment4>
```

```
> cd "g:\C\Assignment4\" ; if ($?) { gcc Q2Palindrom.c -o Q2Palindrom } ; if ($?) { .\Q2Palindrom }
Enter an integer: 121
121 is a palindrome.
PS G:\C\Assignment4>
```

C > Assignment4 > C Q3Armstrong.c > ...

```c
// Check Number is Armstrong or not
#include <math.h>
#include <stdio.h>

int main() {
    int num, originalNum, remainder, n = 0;
    float result = 0.0;

    printf("Enter an integer: ");
    scanf("%d", &num);

    originalNum = num;

    // store the number of digits of num in n
    for (originalNum = num; originalNum != 0; ++n) {
        originalNum /= 10;
    }

    for (originalNum = num; originalNum != 0; originalNum /= 10) {
        remainder = originalNum % 10;

        // store the sum of the power of individual digits in result
        result += pow(remainder, n);
    }

    // if num is equal to result, the number is an Armstrong number
    if ((int)result == num)
     printf("%d is an Armstrong number.", num);
    else
     printf("%d is not an Armstrong number.", num);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
> cd "g:\C\Assignment4\" ; if ($?) { gcc Q3Armstrong.c -o Q3Armstrong } ; if ($?) { .\Q3Armstrong }
Enter an integer: 153
153 is an Armstrong number.
```

C: > Users > Asus > Desktop > C the_series_q4.c > main()

```c
#include<stdio.h>

int main(){
    int n;
    scanf("%d",&n);
    int sum = 0;
    for (int i = 1 ; i <= n ; i++){
        for (int j = 1 ; j <= i ; j++){
            sum += j;
        }
    }
    printf("the sum is %d",sum);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS G:\C> cd "c:\Users\Asus\Desktop\" ; if ($?) { gcc the_series_q4.c -o the_series_q4 } ; if ($?) { .\the_series_q4 }
45
the sum is 16215
```

C > Assignment4 > C Q5Evaluate_Series.c > main()

```c
//Evaluate Series 1-x+(x2/2!)- (x3/3!)+ (x4/4!)+….n term
#include<stdio.h>
#include<conio.h>
#include<math.h>
    void main()
    {
        long int n, s=1 , f=1, i , sign = 1 , x,j;
         printf ("Enter number of terms\n");
         scanf ("%d",&n);
         printf ("Enter value of x\n");
         scanf ("%d",&x);
         for ( i=1 ; i <= n ; i++)
          {
             for ( j=1 ; j<=i ; j++)
             {
                 f = f*i;

             }
             sign = sign*(-1);
             s = s - sign * pow(x,i)/f;
             f = 1;
          }
        printf ("%d is a sum of series",s);
        getch();
    }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
5
Enter value of x
6
2 is a sum of series
```

C > Assignment4 > C Q6Unary_Operator.c > main()

```c
//Evaluate a =a+2 using unary operator
#include <stdio.h>

int main () {
    int a=1;
    printf("%d\n",++a);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS G:\C\Assignment4> cd "g:\C\Assignment4\" ; if ($?) { gcc Q6Unary_Operator.c -o Q6Unary_Operator } ; if ($?) { .\Q6Unary_Operator }
2
```

C > Assignment4 > C Q7Add_1st_3rd_of_3digitnum.c > ...

```c
// Addition of first-digit and last-digit of a three digit number
#include <stdio.h>
int firstDigit(int n)
{
    //Remove last digit from number
    //till only one digit is left
    while (n >= 10)
        n /= 10;
    //Return the first digit
    return n;
}
int lastDigit(int n)
{
    //Return the last digit
    return (n % 10);
}
int main()
{
    int n, sum = 0, firstDigit, lastDigit,digit;
    printf("Enter number to find sum of first and last digit = ");
    scanf("%d", &n);
    //Find last digit of a number
    lastDigit = n % 10;
    //Find total number of digit - 1
    digit = (int)log10(n);
    //Find first digit
    firstDigit = (int) (n / pow(10, digit));
    //Calculate sum of first and last digit
    sum = firstDigit + lastDigit;
    printf("Sum of first and last digit = %d", sum);
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Enter number to find sum of first and last digit = 123
Sum of first and last digit = 4
```

C > Assignment4 > C Q8(i)Swap_Twonum.c > main()

```c
//By using bit wise operator " ^ "
#include <stdio.h>
int main()
{
    int x = 10, y = 5;


    x = x ^ y;
    y = x ^ y;
    x = x ^ y;

    printf("After Swapping: x = %d, y = %d", x, y);

    return 0;
}
```

```c
//By using arithmetic operators “+, -”
#include <stdio.h>
int main()
{
    int x = 10, y = 5;


    x = x + y;
    y = x - y;
    x = x - y;

    printf("After Swapping: x = %d, y = %d", x, y);

    return 0;
}
```

C > Assignment4 > C Q9Print_Special_Symbol.c > main()

```c
//. Read and print special characters and symbols from keyboard.
#include<stdio.h>

int main()
{
    char ch;

    printf("Enter a Special Character\n");
    scanf("%c", &ch);


    if( (ch >= 0  && ch <= 47) ||
        (ch >= 58 && ch <= 64) ||
        (ch >= 91 && ch <= 96) ||
        (ch >= 123 && ch <= 127))
    {
        printf("%c is a Special Character\n", ch);
    }
    else
    {
        printf("Enter the Special Character and Symbol Only");
    }
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
                > cd "g:\C\Assignment4\" ; if ($?) { gcc Q9Print_Special_Symbol.
Enter a Special Character
=
= is a Special Character
PS G:\C\Assignment4>
```