1.Binary Representation

.vscode > C binary.c

```c
#include<stdio.h>
int main() {
    int dec,binary[32],index=0;
    printf("To convert decimal number to binary\n\n");
    printf("Enter a decimal number\n");
    scanf("%d",&dec);
    if(dec == 0) {
        printf("Binary Representation: 0");
    }
    while(dec>0) {
        binary[index]=dec%2;
        dec/=2;
        index++;
    }
    printf("Binary Representation: ");
    for(int i=index-1;i>=0;i--) {
        printf("%d",binary[i]);
    }
    printf("\n\n");
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output> & .\'binary.exe'
To convert decimal number to binary

Enter a decimal number
5
Binary Representation: 101
```

2.Count Vowels and Consonants in String using Pointers

.vscode > C vowels.c

```c
#include<stdio.h>
int main() {
    char str[100],*ptr;
    int vow=0,con=0;
    printf("\nTo count vowels and consonants in string using pointer\n\n");
    printf("Enter a string: ");
    fgets(str,sizeof(str),stdin);
    ptr=str;
    while(*ptr!='\0') {
        if((*ptr>='A'&& *ptr<='z')||(*ptr>='a'&&*ptr<='z')) {
            if(*ptr=='A'||*ptr=='E'||*ptr=='I'||*ptr=='O'||*ptr=='U'||*ptr=='a'||*ptr=='e'||*ptr=='i'||*ptr=='o'||*ptr=='u') {
                vow++;
            } else {
                con++;
            }
        }
        ptr++;
    }
    printf("Number of vowels:%d\n",vow);
    printf("Number of consonants:%d\n",con);
    printf("\n\n");
    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output'
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output> & .\'vowels.exe'

To count vowels and consonants in string using pointer

Enter a string: Hello World
Number of vowels:3
Number of consonants:7
```

3.Functions(Menu-Driven Program)

```c
#include<stdio.h>
void add(int a,int b);
void subtract(int a, int b);
void multiply(int a,int b);
void divide(int a,int b);
int main(){
  int a,b;
  int choice;
  while(1){
  printf("MENU\n1. ADDITION\n2. SUBTRACTION\n3. MULTIPLICATION\n4. DIVISION\n5. EXIT\n");
  printf("Enter your choice: ");
  scanf("%d",&choice);
  if (choice == 5) {
    printf("Exiting the program\n");
    break;
  }
  printf("Enter two numbers : ");
  scanf("%d %d",&a,&b);
  switch(choice){
    case 1: add(a,b);
    break;
    case 2: subtract(a,b);
    break;
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output\output'
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output\output> & .\'arithmeticfunctions.exe'
MENU
1. ADDITION
2. SUBTRACTION
3. MULTIPLICATION
4. DIVISION
5. EXIT
Enter your choice: 2
Enter two numbers : 5 3
Difference of a and b : 2
```

```c
  switch(choice){
    case 1: add(a,b);
    break;
    case 2: subtract(a,b);
    break;
    case 3: multiply(a,b);
    break;
    case 4: divide(a,b);
    break;
    default:
    printf("Enter a proper choice\n");
  }

}
}
void add(int a,int b){
  printf("Sum of a and b : %d\n",a+b);
}
void subtract(int a,int b){
  printf("Difference of a and b : %d\n",a-b);
}
void multiply(int a, int b){
  printf("a x b : %d\n",a*b);
}
void divide(int a,int b){
  printf("a / b : %d\n",a/b);
}
```

4.Diamond Shaped Pattern

.vscode > C diamondpattern.c

```c
#include<stdio.h>
int main() {
    int n=5;
    int i,j,k;
    for(i=1;i<=5;i++) {
        for(j=1;j<=5-i;j++) {
            printf(" ");
        }
        for(k=1;k<=2*i-1;k++) {
            printf("*");
        }
        printf("\n");
    }
    for(i=4;i>=1;i--) {
        for(j=1;j<=5-i;j++) {
            printf(" ");
        }
        for(k=1;k<=2*i-1;k++) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output'
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output> & .\'diamondpattern.exe'
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

5.Length of String using Pointers

.vscode > C stringlength.c

```c
#include<stdio.h>
#include<string.h>
int main() {
    printf("\nTo find length of the string using pointers\n\n");
    char str[]="Hello";
    char *ptr=str;
    int length;
    length=strlen(ptr);
    printf("Length of the string is %d\n",length);
    printf("\n\n");
    return 0;
}
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output'
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output> & .\'stringlength.exe'

To find length of the string using pointers

Length of the string is 5
```

6.GCD using Recursion

.vscode > C recursion(gcd).c

```c
#include<stdio.h>
int gcd(int,int);
int main() {
    int a,b;
    printf("Enter two numbers: \n");
    scanf("%d %d",&a,&b);
    printf("GCD is %d",gcd(a,b));
    return 0;
}
int gcd(int x,int y) {
    if(x==0)
        return y;
    else if(y==0)
        return x;
    else
        return gcd(y,x % y);
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode> cd 'c:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output'
PS C:\Users\abhin\OneDrive\Desktop\C programs\.vscode\.vscode\output> & .\'recursion(gcd).exe'
Enter two numbers:
56 98
GCD is 14
```