Assignment – 1 (Control Statements and Loops)

1. Write a C program to check positive, negative or zero using simple if or if else. C program to input any number from user and check whether the given number is positive, negative or zero. Logic to check negative, positive or zero in C programming.

   Example
   Input
   Input number: 23
   Output
   23 is positive

2. Write a C program to check whether a number is divisible by 5 and 11 or not using if else. How to check divisibility of any number in C programming. C program to enter any number and check whether it is divisible by 5 and 11 or not. Logic to check divisibility of a number in C program.

   Example
   Input
   Input number: 55
   Output
   Number is divisible by 5 and 11

3. Write a C program to input a character from user and check whether the given character is alphabet or not using if else. How to check whether a character is alphabet or not in C programming. Logic to check if a character is alphabet or not in C program.

   Example
   Input
   Input character: a
   Output
   'a' is alphabet

4. Write a C program to count the Vowels in the given string.

5. Write a C program to input character from user and check whether character is uppercase or lowercase alphabet using if else. How to check uppercase and lowercase using if else in C programming. Logic to check uppercase and lowercase alphabets in C program.
   Example
   Input
   Input character: C
   Output
   'C' is uppercase alphabet

6. Write a C program to input amount from user and print minimum number of notes (Rs. 500, 100, 50, 20, 10, 5, 2, 1) required for the amount. How to the minimum number of notes required for the given amount in C programming. Program to find minimum number of notes required for the given denomination. Logic to find minimum number of denomination for a given amount in C program.

Example
Input
Input amount: 575
Output
Total number of notes:
500: 1
100: 0
50: 1
20: 1
10: 0
5: 1
2: 0
1: 0

7. Write a C program to input a number from user and count number of digits in the given integer using loop. How to find total digits in a given integer using loop in C programming. Logic to count digits in a given integer without using loop in C program.

   Example
   Input
   Input num: 35419
   Output
   Number of digits: 5

8. Write a C program to input a number and calculate sum of digits using for loop. How to find sum of digits of a number in C program. Logic to find sum of digits of a given number in C programming.

   Example
   Input
   Input any number: 1234
   Output
   Sum of digits: 10

9. Write a C program to input a number from user and find reverse of the given number using for loop. How to find reverse of any number using loop in C program. Logic to find reverse of a number in C programming.

   Example
   Input
   Input number: 12345
   Output
   Reverse of 12345 = 54321

10. Write a C program to input decimal number from user and convert to binary number system. How to convert from decimal number to binary number system in C program. Logic to convert decimal to binary number system in C programming.

    Example
    Input
    Input decimal number: 112
    Output
    Binary number: 0111000

main.c

Run

Output

Clear

```c
#include <stdio.h>
int main()
{
    int number;

    printf("Enter a number: ");
    scanf("%d", &number);

    if (number > 0) {
        printf("%d is positive", number);
    }
    else if (number < 0) {
        printf("%d is negative", number);
    }
    else {
        printf("The number is zero");
    }

return 0;
}
```

```
/tmp/OmPeCipqSE.o
Enter a number: 23
23 is positive
```

main.c

Run

```c
#include <stdio.h>

int main() {
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);

    if(num % 5 == 0 && num % 11 == 0) {
        printf("Number is divisible by 5 and 11");
    }
    else {
        printf("Number is not divisible by 5 and 11");
    }

return 0;
}
```

Output

Clear

```
/tmp/OmPeCipqSE.o
Enter a number: 55
Number is divisible by 5 and 11
```

main.c

Run

Output

Clear

```c
#include <stdio.h>

int main() {
    char ch;
    printf("Input character: ");
    scanf("%c", &ch);

    if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
        printf("'%c' is alphabet\n", ch);
    } else {
        printf("'%c' is not alphabet\n", ch);
    }
return 0;
}
```

```
/tmp/OmPeCipqSE.o
Input character: a
'a' is alphabet
```

main.c

Run

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

int main() {
    char str[100];
    int count = 0, i;

    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);

    for(i = 0; i < strlen(str); i++) {
        if(str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' ||
            str[i] == 'u' ||
            str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' ||
                str[i] == 'U') {
            count++;
        }
    }

    printf("Number of vowels in the string: %d\n", count);
return 0;
}
```

Output

Clear

```
/tmp/OmPeCipqSE.o
Enter a string: elephant
Number of vowels in the string: 3
```

main.c

Run

Output

Clear

```c
#include <stdio.h>

int main() {
   char ch;

   printf("Enter a character: ");
   scanf("%c", &ch);

   if (ch >= 'A' && ch <= 'Z')
      printf("%c is an uppercase alphabet.", ch);
   else if (ch >= 'a' && ch <= 'z')
      printf("%c is a lowercase alphabet.", ch);
   else
      printf("%c is not an alphabet.", ch);

return 0;
}
```

```
/tmp/OmPeCipqSE.o
Enter a character: C
C is an uppercase alphabet.
```

main.c

Run

```c
#include <stdio.h>

int main()
{
    int amount, notes;
    int denominations[] = { 500, 100, 50, 20, 10, 5, 2, 1 };
    int noteCount[sizeof(denominations)/sizeof(denominations[0])];
    int i;

    // Input amount
    printf("Enter amount: ");
    scanf("%d", &amount);

    // Initialize note count for all denominations to zero
    for(i = 0; i < sizeof(denominations)/sizeof(denominations[0]); i++) {
        noteCount[i] = 0;
    }

    // Find minimum number of notes for each denomination
    for(i = 0; i < sizeof(denominations)/sizeof(denominations[0]); i++) {
        if(amount >= denominations[i]) {
            notes = amount / denominations[i];
            noteCount[i] = notes;
            amount -= notes * denominations[i];
        }
```

Output

Clear

```
/tmp/0mPeCipqSE.o
Enter amount: 575
Total number of notes:
500: 1
50: 1
20: 1
5: 1
```

main.c

Run

```c
#include <stdio.h>

int main() {
    int num, count = 0;

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

    while (num != 0) {
        count++;
        num /= 10;
    }

    printf("Number of digits: %d\n", count);

return 0;
}
```

Output

Clear

```
/tmp/OmPeCipqSE.o
Enter an integer: 35419
Number of digits: 5
```

main.c

Run

Output

Clear

```c
#include <stdio.h>

int main() {
    int num, sum=0, digit;

    printf("Enter a number: ");
    scanf("%d", &num);

    // calculate sum of digits using a for loop
    for(;num>0;num/=10) {
        digit = num % 10;
        sum += digit;
    }

    printf("Sum of digits: %d", sum);

return 0;
}
```

```
/tmp/OmPeCipqSE.o
Enter a number: 1234
Sum of digits: 10
```

main.c

Run

```c
#include <stdio.h>

int main() {
    int num, rev_num = 0, rem;

    printf("Enter a number: ");
    scanf("%d", &num);

    // loop to reverse the number
    for(; num != 0; num /= 10) {
        rem = num % 10;
        rev_num = rev_num * 10 + rem;
    }

    printf("Reverse of %d = %d", num, rev_num);

return 0;
}
```

Output

Clear

```
/tmp/OmPeCipqSE.o
Enter a number: 12345
Reverse of 0 = 54321
```

main.c

Run

Output

Clear

```c
#include <stdio.h>

int main() {
    int decimal_num, binary_num[20], i = 0;

    printf("Enter a decimal number: ");
    scanf("%d", &decimal_num);

    while (decimal_num > 0) {
        binary_num[i] = decimal_num % 2;
        decimal_num = decimal_num / 2;
        i++;
    }

    printf("Binary number: ");
    for (int j = i - 1; j >= 0; j--)
        printf("%d", binary_num[j]);

return 0;
}
```

```
/tmp/OmPeCipqSE.o
Enter a decimal number: 112
Binary number: 1110000
```