MODULE: 3.2 (C Language PrograMODULE: 3.2 (C Language Programing with C)ming with C)

1. WAP to make simple calculator (operation include Addition, Subtraction, Multiplication, Division, modulo)

Code:

```c
#include <stdio.h>

int main(void){
   // taking operator as character from the user and storing in op
   char op;
   // taking two numbers input and storing in the n1,n2
   int n1,n2;

   printf("Enter operator (+,-,/,*)");
   scanf("%c",&op);

   printf("Enter two numbers: ");
   scanf("%d %d",&n1,&n2);

// To add
   if(op == '+'){
       printf("Addition: %d\n",n1+n2);
   }

// To subtract
   else if (op == '-'){
       printf("Subtraction: %d\n",n1-n2);
   }
// To multiply
   else if(op == '*'){
       printf("Multiplication: %d\n",n1*n2);
   }
// To divide

   else if(op == '/'){
       printf("Division is %d\n",n1/n2);
   }
// To  modulo
```

```
    else{
        printf("Modulo is: %d\n",n1%n2);
    }

    return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/first
Enter operator (+,-,/,*)*
Enter two numbers: 10 40
Multiplication: 400
```

2. WAP to swap two numbers without using third variable

Code:

```
#include <stdio.h>



int main(void){
    int a,b;
    a = 20;
    b = 40;
    printf("Before: A=%d, B=%d\n",a,b);

    // First all a + b is stored in a
    a = a+b;
    // a is subtracted by b so b gets the value of a
    b = a-b;
    // b is subtracted by the value of a;
    a = a-b;
    printf("After: A=%d, B=%d\n",a,b);

    return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments
└──╼ $maker module_3_2/second
Before: A=20, B=40
After: A=40, B=20
```

3. WAP to find number is even or odd using ternary operator

Code:

```c
#include <stdio.h>

int main() {
   int number;

   // Input a number from the user
   printf("Enter a number: ");
   scanf("%d", &number);

   // Check if the number is even or odd using the ternary operator
   (number % 2 == 0) ? printf("%d is even.\n", number) : printf("%d is
odd.\n", number);

   return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/third
Enter a number: 20
20 is even.
```

4. WAP to show
   1. Monday to Sunday using switch case
   2. Vowel or Consonant using switch case

Code:

```c
// WAP to show
// 1. Monday to Sunday using switch case
// 2. Vowel or Consonant using switch case
```

```c
#include <stdio.h>


int main(void){
   int day;
      // take day numbers as an input
   printf("Enter week day: ");
   scanf("%d",&day);

   switch(day){
      // if day == 1 then tis' monday
       case 1:
           printf("Monday\n");
           Break;
      // if day == 2 then tis' tuesday
       case 2:
           printf("Tuesday\n");
           break;
       case 3:
      // if day == 3 then tis' wednesday
           printf("Wednesday\n");
           Break;
      // so on…
       case 4:
           printf("Thursday\n");
           break;
       case 5:
           printf("Friday\n");
           break;
       case 6:
           printf("Saturday\n");
           break;
       case 7:
           printf("Sunday\n");
           Break;
      // if none of the conditions above satisfies then tis' invalid
       default:
           printf("Invalid choice\n");
   }
```

```
    return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/fourth
Enter week day: 3
Wednesday
```

```c
#include <stdio.h>


int main(void){
    char a;
    // Take a character input from the user
    printf("Enter char: ");
    scanf("%c",&a);
    // if the character is a capital letter then convert it to lowercase by adding
32 to it's ascii value
    if (a < 90){
        a = a+ 32;
    }

    // Compare a
    switch(a){
        // If case is a then it's vowel
        case 'a':
            printf("Vowel\n");
            break;
        // If case is b then it's vowel
        case 'e':
            printf("Vowel\n");
            break;
        // So on....
        case 'i':
            printf("Vowel\n");
            break;

        case 'o':
            printf("Vowel\n");
            break;

        case 'u':
```

```
            printf("Vowel\n");
            break;

        // otherwise it's a consonant

        default:
            printf("Consonant\n");
            break;
    }

    return 0;
}
```

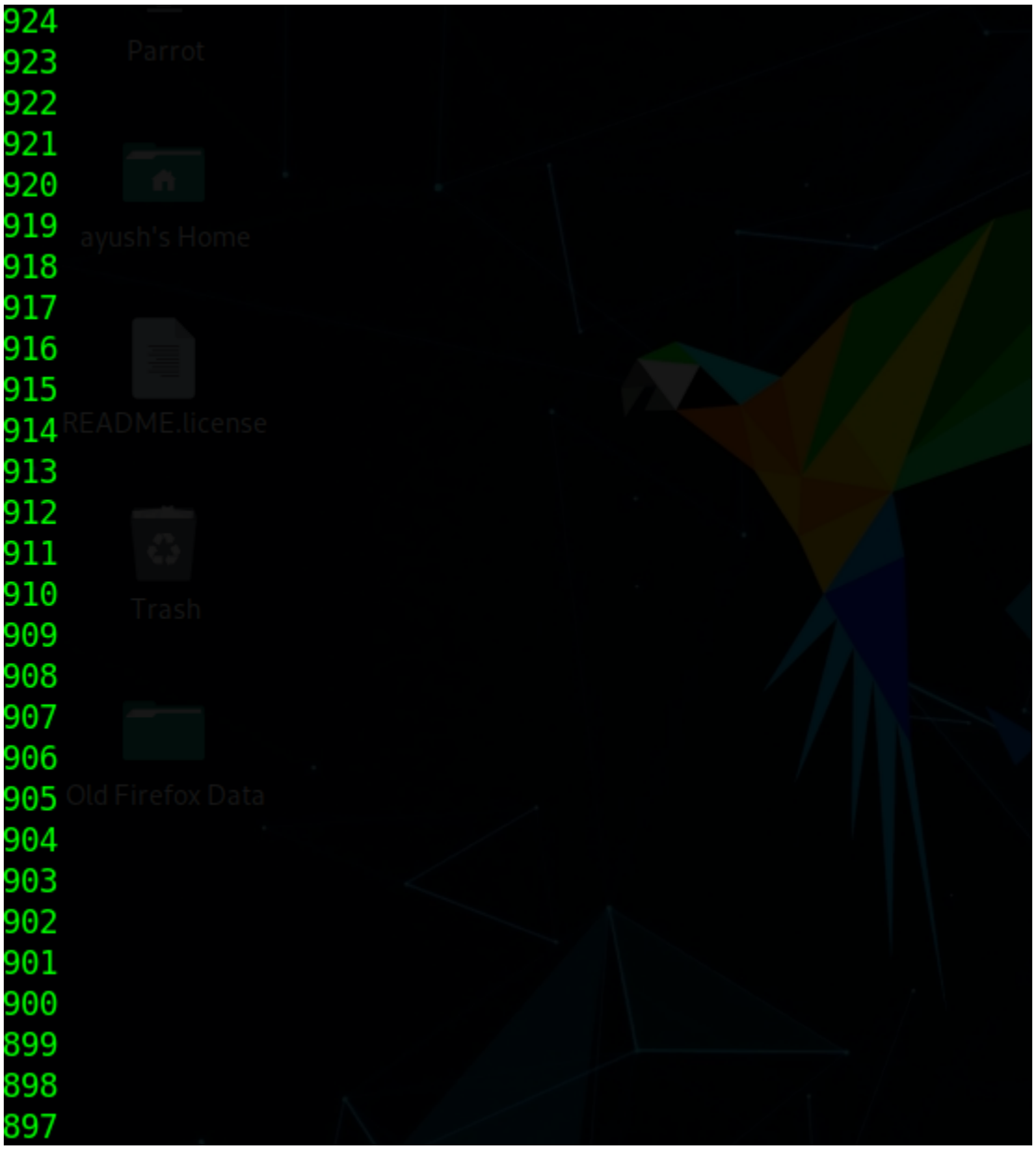
```
┌─[ayush@security]-[~/c/assignments]
└──╼ $maker module_3_2/four_2
Enter char: A
Vowel
┌─[ayush@security]-[~/c/assignments]
└──╼ $maker module_3_2/four_2
Enter char: E
Vowel
┌─[ayush@security]-[~/c/assignments]
└──╼ $maker module_3_2/four_2
Enter char: e
Vowel
```

5. WAP to print 972 to 897 using for loop

Code:

```
#include <stdio.h>


int main(void){

    // assign the value of i as 972
    // then compare if i is greater than and equals to 897
    // if its greater than decrease the value of i by 1;
    // if it's smaller than limit than stop the loop
    for(int i = 972;i>=897;i--){

        printf("%d\n",i);
```

```
    }


    return 0;
}
```



6. WAP to take 10 no. Input from user and find out how many even numbers are there.

Code:

```
#include <stdio.h>


int main(void){
```

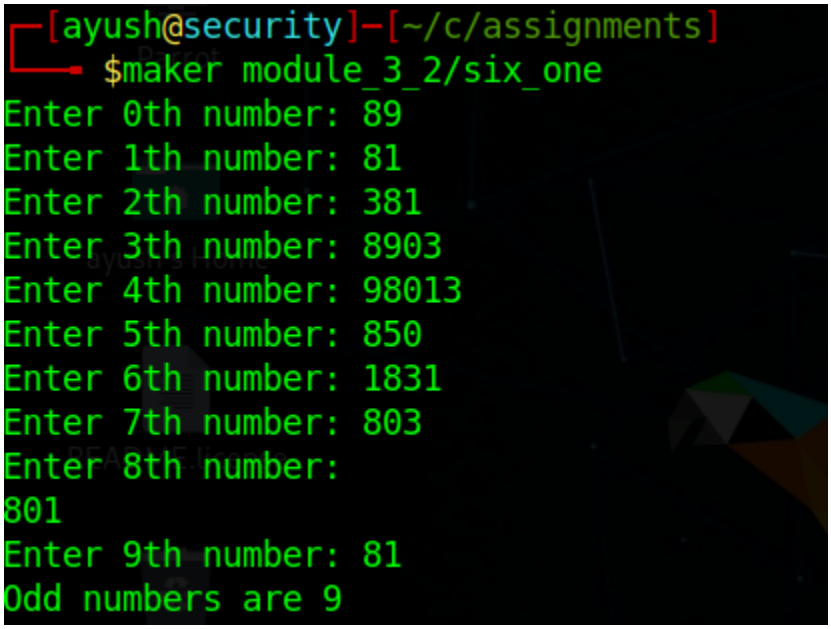
```c
   // Store user input into nums, and initilize the even number counter to
zero
   int nums,evenNums=0;

   for(int i =0;i<10;i++){
       // Ask user for 10 inputs
       printf("Enter %dth number: ",i);
       scanf("%d",&nums);

       // if the the modulo of nums is 0 then increment the counter by 1
       if(nums%2==0){

           evenNums+=1;
       }

   }
   // Display the even number counter
   printf("Even numbers are %d\n",evenNums);
   return 0;
}
```

- How many odd numbers are there

Code:

```c
#include <stdio.h>

int main(void){
```

```c
   // Store user input into nums, and initilize the odd number counter to
zero
   int nums,oddNums=0;

   for(int i =0;i<10;i++){
       // Ask user for 10 inputs
       printf("Enter %dth number: ",i);
       scanf("%d",&nums);

       // if the the modulo of nums is 0 then increment the counter by 1
       if(nums%2!=0){

           oddNums+=1;
       }

   }
   // Display the odd number counter
   printf("Odd numbers are %d\n",oddNums);
   return 0;
}
```



- Sum of even numbers

Code

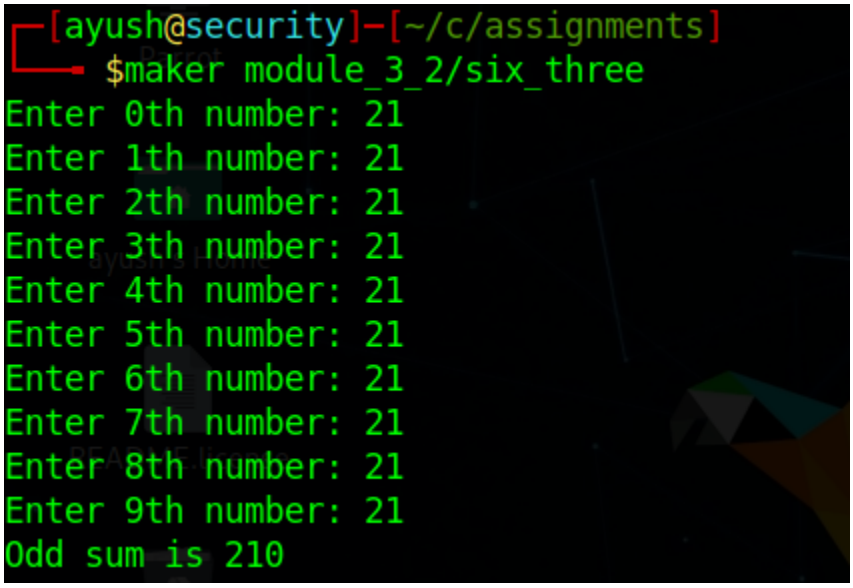
```c
#include <stdio.h>

int main(void){
   // Store user input into nums, and initilize the evenSum counter to
zero
   int nums,evenSum=0;

   for(int i =0;i<10;i++){
       // Ask user for 10 inputs
       printf("Enter %dth number: ",i);
       scanf("%d",&nums);

       // if the the modulo of nums is 0 then add the sum by the nums
       if(nums%2==0){

           evenSum+=nums;
       }

   }
   // Display the even sum
   printf("Even sum is %d\n",evenSum);
   return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/six_two
Enter 0th number: 10
Enter 1th number: 20
Enter 2th number: 30
Enter 3th number: 40
Enter 4th number: 50
Enter 5th number: 60
Enter 6th number: 70
Enter 7th number: 80
Enter 8th number: 80
Enter 9th number: 100
Even sum is 540
```

- Sum of odd numbers

Code:

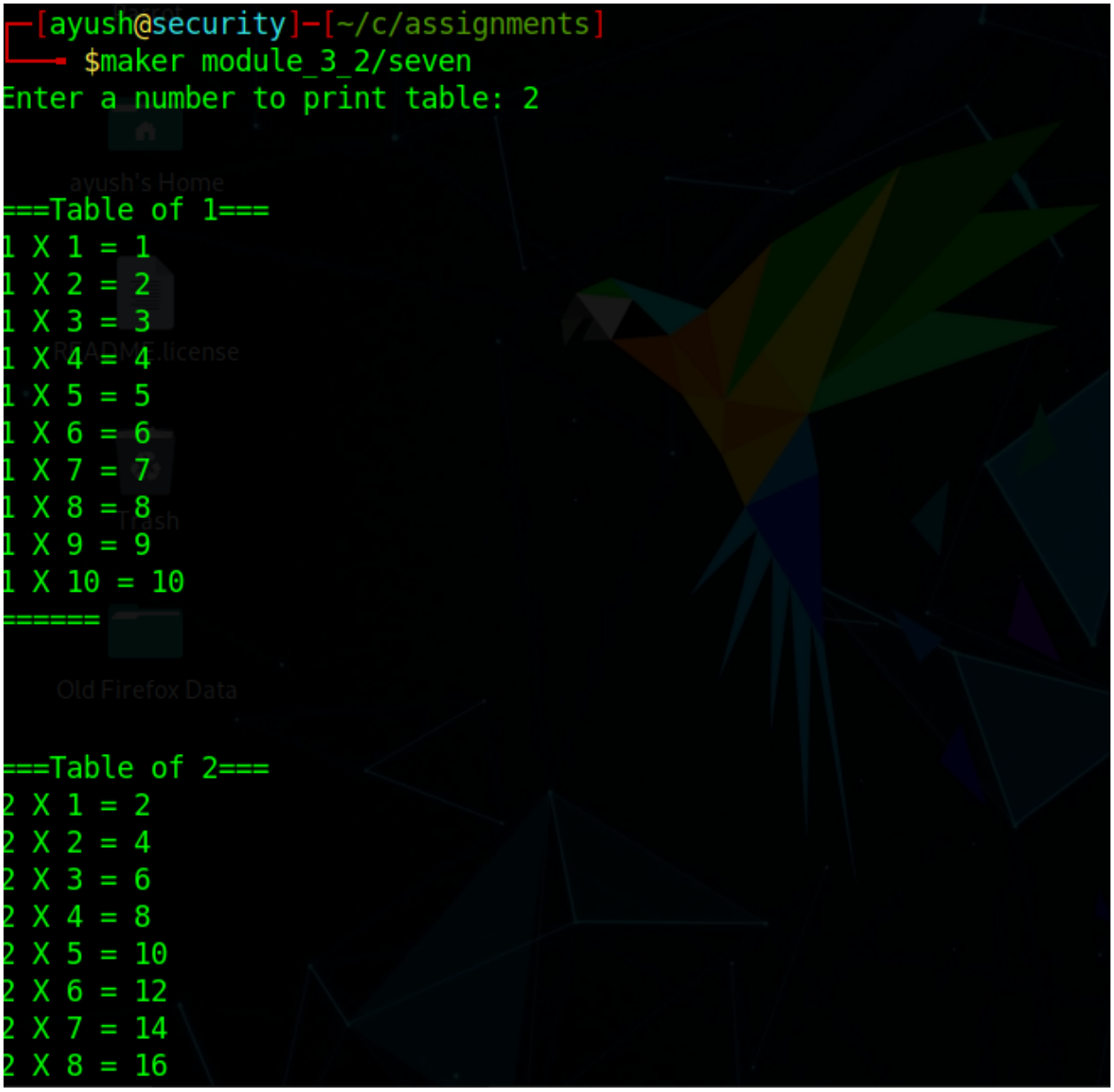
```c
#include <stdio.h>

int main(void){
   // Store user input into nums, and initilize the oddSum counter to zero
   int nums,oddSum=0;

   for(int i =0;i<10;i++){
       // Ask user for 10 inputs
       printf("Enter %dth number: ",i);
       scanf("%d",&nums);

       // if the the modulo of nums is 0 then add the sum by the nums
       if(nums%2!=0){

           oddSum+=nums;
       }

   }
   // Display the even sum
   printf("Odd sum is %d\n",oddSum);
   return 0;
}
```



- WAP to print table up to given numbers

Code:

```c
#include <stdio.h>

int main(void){
// Store the user input in tableN
   int tableN;

//  Ask the user to input a number
   printf("Enter a number to print table: ");
   scanf("%d",&tableN);

// Run the outer loop until `i` is less than or equals to tableN print
table of a number

   for(int i=1;i<=tableN;i++){
       printf("\n\n===Table of %d===\n",i);

       // Run the inner to calculate multiplications
       for(int j = 1;j<=10;j++){
           printf("%d X %d = %d\n",i,j,i*j);
       }
       printf("======\n\n");

   }
   return 0;
}
```



- WAP to print factorial of given number

Code:

```c
#include <stdio.h>

// Function prototype for the factorial function
int factorial(int n);

int main(void) {
   // Variable to store user input
   int n;
```

```c
    // Ask the user for input
    printf("Enter a number to find its factorial: ");

    // Read user input
    scanf("%d", &n);

    // Display the factorial of the entered number
    printf("Factorial: %d\n", factorial(n));

    return 0;
}

// Function to calculate the factorial of a number
int factorial(int n) {
    // Variable to store the result
    int result = 1;

    // Calculate factorial using a loop
    for (int i = n; i >= 1; i--) {
        result = result * i;
    }

    // Return the result
    return result;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/eight
Enter a number to find its factorial: 5
Factorial: 120
```

- WAP to print Fibonacci series up to given numbers

Code:

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

```c
int main(void) {
   // Initialize variables for the Fibonacci sequence
   int first = 0, second = 1, next;

   // Set the number of terms in the sequence
   int n = 10;

   // Iterate to generate the Fibonacci sequence
   for (int i = 0; i < n; i++) {
       // Update values in the sequence
       first = second;
       second = next;
       next = first + second;
   }

   // Display the nth term of the Fibonacci sequence
   printf("The %dth term of the Fibonacci sequence: %d\n", n, next);

   return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/nine
The 10th term of the Fibonacci sequence: 55
```

- WAP to print number in reverse order e.g.: number = 64728 ---> reverse = 82746

Code:

```c
#include <stdio.h>

int main(void) {
   // Variables to store user input, remainder, and the reversed number
   int n, remainder, result = 0;

   // Prompt the user to enter a number for reversal
```

```c
    printf("Enter a number to reverse: ");

    // Read the user input
    scanf("%d", &n);

    // Display the reversed number
    printf("Reverse is: ");

    // Loop to reverse the number
    while (n != 0) {
        // Extract the last digit (remainder)
        remainder = n % 10;

        // Build the reversed number by adding the remainder
        result = result * 10 + remainder;

        // Remove the last digit from the original number
        n /= 10;

        // Print the reversed digit
        printf("%d", remainder);
    }

    // Display the reversed number
    printf("\n");

    return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/ten
Enter a number to reverse: 890
Reverse is : 098
```

- Write a program to find out the max from given number (E.g., No: -1562
  Max number is 6)

Code:

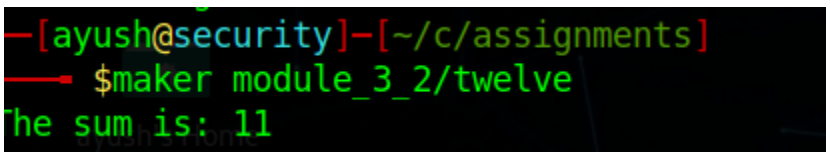
```c
#include <stdio.h>

int main(void) {
   // Variables to store the original number, maximum digit, and remainder
   int n, max = 0, remainder;

   // Set the initial value of n (original number)
   n = 1569;

   // Loop to find the maximum digit in the number
   while (n != 0) {
       // Extract the last digit (remainder)
       remainder = n % 10;

       // Check if the current digit is greater than the current maximum
       if (remainder > max) {
           // Update the maximum digit
           max = remainder;
       }

       // Remove the last digit from the original number
       n /= 10;
   }

   // Display the maximum digit
   printf("The max digit is: %d\n", max);

   return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/eleven
The max digit is: 9
```

- Write a program make a summation of given number (E.g., 1523 Ans: -11)

Code:

```c
#include <stdio.h>

int main(void) {
   // Variables to store the original number, sum of digits, and remainder
   int n, result = 0, remainder;

   // Set the initial value of n (original number)
   n = 1523;

   // Loop to calculate the sum of digits in the number
   while (n != 0) {
       // Extract the last digit (remainder)
       remainder = n % 10;

       // Add the current digit to the running sum
       result += remainder;

       // Remove the last digit from the original number
       n /= 10;
   }

   // Display the sum of digits
   printf("The sum of digits is: %d\n", result);

   return 0;
}
```



- Write a program you have to make a summation of first and last Digit. (E.g., 1234 Ans: -5)

Code:

```c
#include <stdio.h>

int main(void) {
   // Variables to store the original number, remainder, result, and a temporary counter
   int n = 1234, remainder, result = 0, temp = 1;

   // Loop to iterate through the digits of the number
   while (n != 0) {
       // Extract the last digit (remainder)
       remainder = n % 10;

       // Check if it is the first digit and add it to the result
       if (temp == 1) {
           result += remainder;
       }

       // Remove the last digit from the original number
       n /= 10;

       // Increment the temporary counter
       temp += 1;
   }

   // Add the last digit to the result
   result += remainder;

   // Display the sum of the first and last digits
   printf("The sum of the first and last digit is: %d\n", result);

   return 0;
}
```

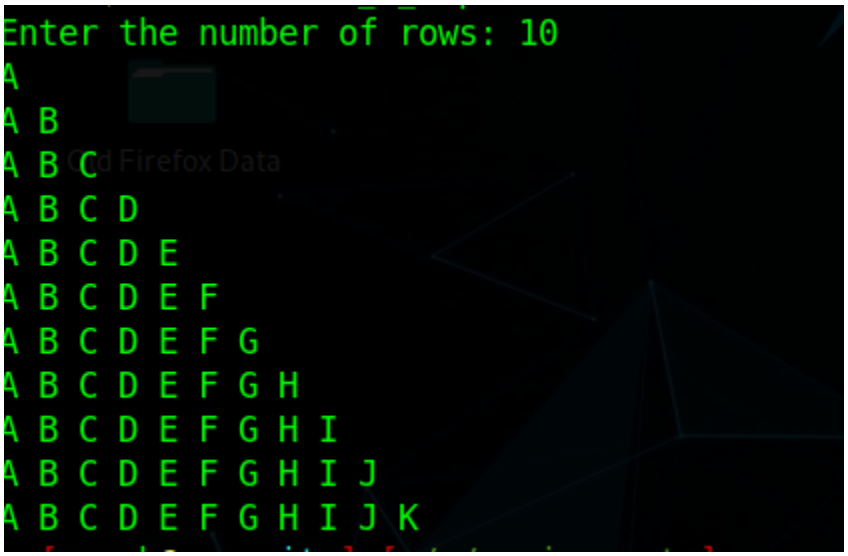
```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/thirteen
The sum of first and last digit is : 5
```

Patterns

```c
#include <stdio.h>

int main(void) {
   // Variable to store the number of rows
   int n;

   // Prompt the user to enter the number of rows
   printf("Enter the number of rows: ");

   // Read the user input
   scanf("%d", &n);

   // Loop to iterate through each row
   for (int i = 0; i < n; i++) {
       // Nested loop to print each element in the row
       for (int j = 0; j <= i; j++) {
           // Check if the column index is even or odd to print '1' or '0'
           if (j % 2 != 0) {
               printf("0 ");
           } else {
               printf("1 ");
           }
       }
       // Move to the next line after printing each row
       printf("\n");
   }

   return 0;
}
```

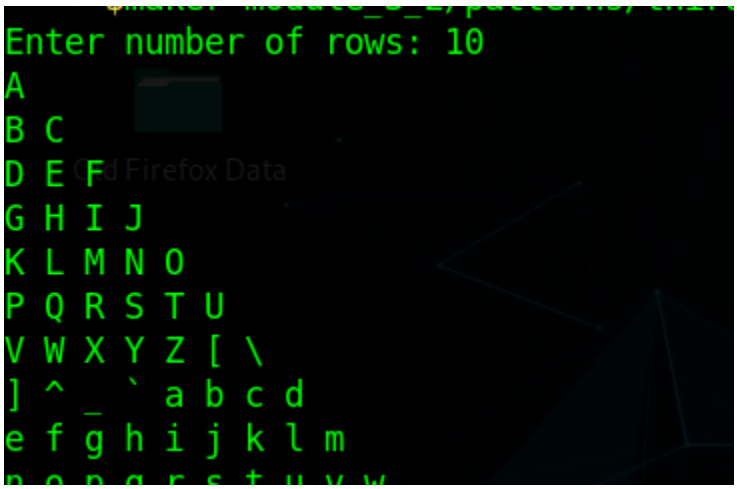
```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/patterns/one
Enter no of rows: 5
A
A 0
A 0 1
A 0 1 0
A 0 1 0 1
```

```c
#include <stdio.h>

int main(void) {
   // Variable to store the number of rows
   int n;

   // Prompt the user to enter the number of rows
   printf("Enter the number of rows: ");

   // Read the user input
   scanf("%d", &n);

   // Loop to iterate through each row
   for (int i = 0; i <= n; i++) {
       // Nested loop to print characters in each row
       for(int k = 'A'; k <= 'A' + i; k++){
           printf("%c ", k);
       }
       // Move to the next line after printing each row
       printf("\n");
   }

   return 0;
}
```



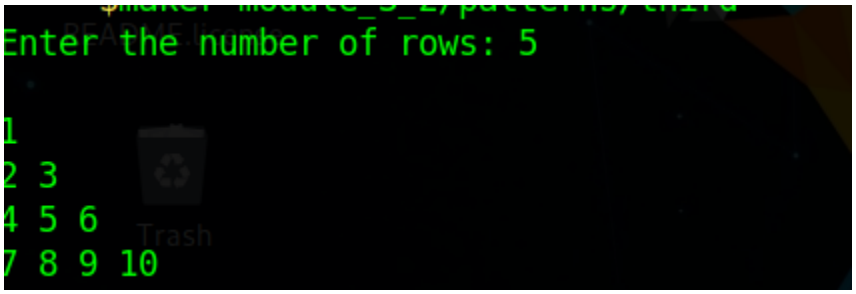

```c
#include <stdio.h>

int main() {
   // Variable to store the number of rows
   int rows;

   // Prompt the user to enter the number of rows
   printf("Enter the number of rows: ");
   scanf("%d", &rows);

   // Variable to store the current alphabet, starting with 'A'
   char n = 'A';

   // Outer loop to print all rows
   for (int i = 0; i < rows; i++) {

       // Inner loop to print alphabet in each row
       for (int j = 0; j <= i; j++) {
           printf("%c ", n++);
       }

       // Move to the next line after printing each row
       printf("\n");
```

```
    }

    return 0;
}
```




```c
#include <stdio.h>

int main() {
    // Variable to store the number of rows
    int rows;

    // Prompt the user to enter the number of rows
    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    // Variable to store the current number, starting with 1
    int n = 1;

    // Outer loop to print all rows
    for (int i = 0; i < rows; i++) {

        // Inner loop to print numbers in each row
        for (int j = 0; j < i; j++) {
            printf("%d ", n++);
        }
```

```
        // Move to the next line after printing each row
        printf("\n");
    }

    return 0;
}
```




```
// C program to print the full pyramid pattern of stars
#include <stdio.h>
 int main()
{
    int rows;
    printf("Enter number of rows: ");
    scanf("%d",&rows);
     // first loop to print all rows
    for (int i = 0; i < rows; i++) {
         // inner loop 1 to print white spaces
        for (int j = 0; j < 2 * (rows - i) - 1; j++) {
            printf(" ");
        }
         // inner loop 2 to print star * character
        for (int k = 0; k < 2 * i + 1; k++) {
            printf("* ");
        }
        printf("\n");
    }
    return 0;
}
```

```
┌─[ayush@security]─[~/c/assignments]
└──╼ $maker module_3_2/patterns/five
Enter number of rows: 5
        *
      * * *
    * * * * *
  * * * * * * *
* * * * * * * * *
```

```c
#include <stdio.h>

int main() {
    // Variable to store the number of rows
    int rows;

    // Prompt the user to enter the number of rows
    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    // Loop to iterate through each row
    for (int i = 1; i <= rows; i++) {

        // Print '*' for increasing number of times in each row
        for (int j = 1; j <= i; j++) {
            printf("*");
        }

        // Move to the next line after printing each row
        printf("\n");
    }

    // Loop to iterate through each row in reverse order, starting from rows-1
    for (int i = rows - 1; i >= 1; i--) {

        // Print '*' for decreasing number of times in each row
        for (int j = 1; j <= i; j++) {
```

```
            printf("*");
        }

        // Move to the next line after printing each row
        printf("\n");
    }

    return 0;
}
```