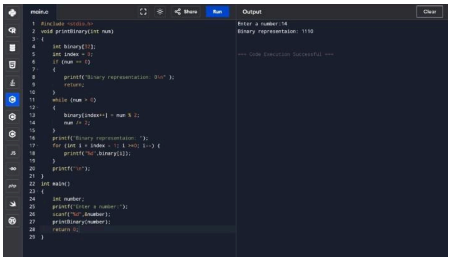
NISSI NAOMI DETA
AP24110012080

LAB ASSIGNMENT-1

1. Write a program to print the binary representation of a number using loops.
Example: Input = 5, Output = 101.

```
#include <stdio.h>
int main() {
int num, binary[32], i = 0;
printf("Enter a number: ");
scanf("%d", &num);
while (num > 0) {
binary[i] = num % 2;
num /= 2;
i++;
}
printf("Binary: ");
for (int j = i - 1; j >= 0; j--) {
printf("%d", binary[j]);
}
return 0;
```

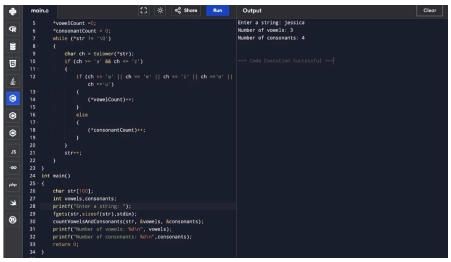


```
}
```

2. Write a program to count the number of vowels and consonants in a string using pointers.

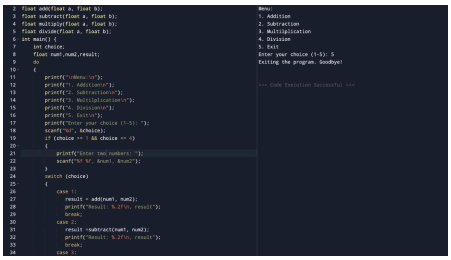
```
#include <stdio.h>
int main() {
char str[100];
int vowels = 0, consonants = 0;
printf("Enter a string: ");
scanf("%s", str);
for (char *ptr = str; *ptr != '\0'; ptr++) {
char ch = *ptr;
if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||
ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {
vowels++;
} else if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
consonants++;
```

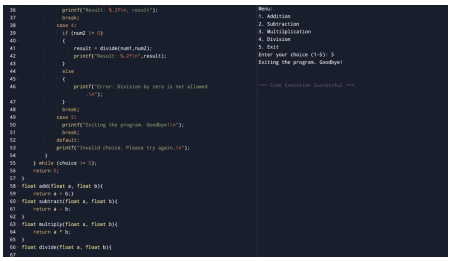
}
}
printf("Vowels: %d, Consonants: %d\n", vowels, consonants);
return 0;
}



3. Write separate functions for addition, subtraction, multiplication, and division.
Call these functions from a menu-driven program.

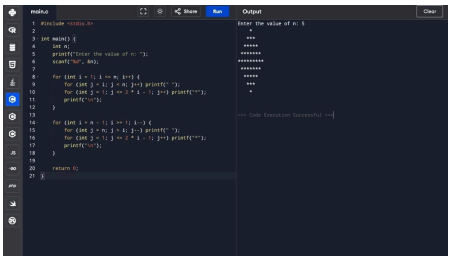
```
#include <stdio.h>
int add(int a, int b) { return a + b; }
int subtract(int a, int b) { return a - b; }
int multiply(int a, int b) { return a * b; }
float divide(int a, int b) { return (float)a / b; }
int main() {
int a, b, choice;
printf("Enter two numbers: ");
scanf("%d %d", &a, &b);
printf("1. Add\n2. Subtract\n3. Multiply\n4. Divide\nEnter choice: ");
scanf("%d", &choice);
switch (choice) {
case 1: printf("Result: %d\n", add(a, b)); break;
case 2: printf("Result: %d\n", subtract(a, b)); break;
case 3: printf("Result: %d\n", multiply(a, b)); break;
case 4: printf("Result: %.2f\n", divide(a, b)); break;
default: printf("Invalid choice!\n");
}
return 0;
```



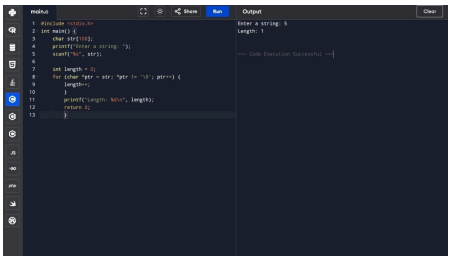


4. Write a program to print a diamond-shaped pattern using loops.
Ex: Input: n = 5
Output:
 *
 ***
 *****
 ****
 ****
 ****
 *****
 ***
 *

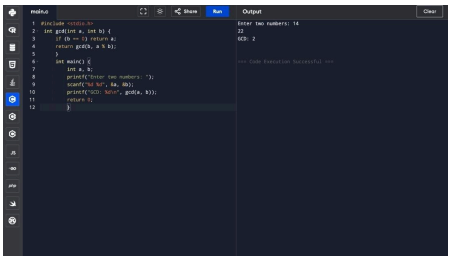
```
#include <stdio.h>
int main() {
int n;
printf("Enter the value of n: ");
scanf("%d", &n);
for (int i = 1; i <= n; i++) {
for (int j = i; j < n; j++) printf(" ");
for (int j = 1; j <= 2 * i - 1; j++) printf("*");
printf("\n");
}
for (int i = n - 1; i >= 1; i--) {
for (int j = n; j > i; j--) printf(" ");
for (int j = 1; j <= 2 * i - 1; j++) printf("*");
printf("\n");
}
return 0;
}
```



5. Write a program to find the length of a given string using pointers.
Ex : Input: "Hello"
Output: 5
#include <stdio.h>
int main() {
char str[100];
printf("Enter a string: ");
scanf("%s", str);
int length = 0;
for (char *ptr = str; *ptr != '\0'; ptr++) {
length++;
}
printf("Length: %d\n", length);
return 0;
}



6.Write a function to calculate the GCD (Greatest Common Divisor) of two
numbers using
recursion.
Ex: Input: a = 56, b = 98
Output: GCD is 14. #include <stdio.h> int gcd(int a, int b) { if (b == 0) return a; return gcd(b, a
% b); }int main() { int a, b; printf("Enter two numb
e
r
s: "
); scanf("%d %d", &a, &b); printf("GCD: %d\n", gcd(a, b)
); return 0; }


main.c
Share
Run
Output
Clear
#include <stdio.h>
int gcd(int a, int b) {
if (b == 0) return a;
return gcd(b, a % b);
}
int main() {
int a, b;
printf("Enter two numbers: ");
scanf("%d %d", &a, &b);
printf("GCD: %d\n", gcd(a, b));
return 0;
}
Enter two numbers: 14
22
GCD: 2
=== Code Execution Successful ===