Computer Network Lab 2

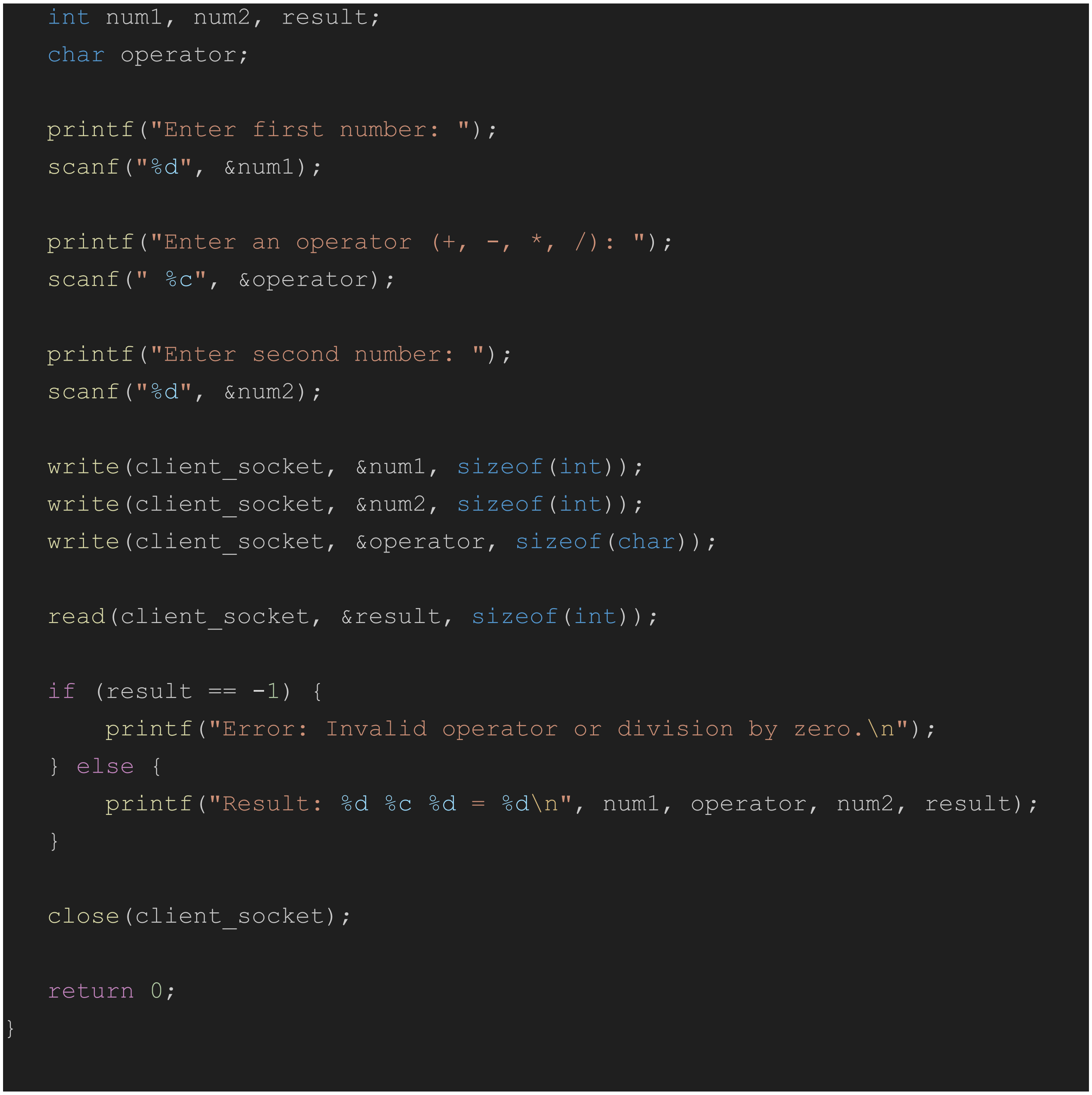
Rajeev Tapadia
IT-C Batch 2
Roll no: 54
PRN: 12210767

- Code:
    - Server Side

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

#define PORT 8080

int calculate(int a, int b, char operator) {
    switch (operator) {
        case '+':
            return a + b;
        case '-':
            return a - b;
        case '*':
            return a * b;
        case '/':
            return b != 0 ? a / b : -1; // Avoid division by zero
        default:
            return -1; // Invalid operator
    }
}

int main() {
    int server_fd, new_socket;
    struct sockaddr_in address;
    int addrlen = sizeof(address);

    if ((server_fd = socket(AF_INET, SOCK_STREAM, 0)) == 0) {
        perror("socket failed");
        exit(EXIT_FAILURE);
    }

    address.sin_family = AF_INET;
    address.sin_addr.s_addr = INADDR_ANY;
    address.sin_port = htons(PORT);

    if (bind(server_fd, (struct sockaddr *)&address, sizeof(address)) < 0) {
        perror("bind failed");
        exit(EXIT_FAILURE);
    }

    if (listen(server_fd, 3) < 0) {
        perror("listen");
        exit(EXIT_FAILURE);
    }

    printf("Server is listening on port %d\n", PORT);

    while (1) {
        if ((new_socket = accept(server_fd, (struct sockaddr *)&address, (socklen_t *)&addrlen)) < 0) {
            perror("accept");
            exit(EXIT_FAILURE);
        }

        int num1, num2, result;
        char operator;

        read(new_socket, &num1, sizeof(int));
        read(new_socket, &num2, sizeof(int));
        read(new_socket, &operator, sizeof(char));

        result = calculate(num1, num2, operator);
        printf("Received: %d %c %d = %d\n", num1, operator, num2, result);

        send(new_socket, &result, sizeof(int), 0);
        close(new_socket);
    }

    return 0;
}
```

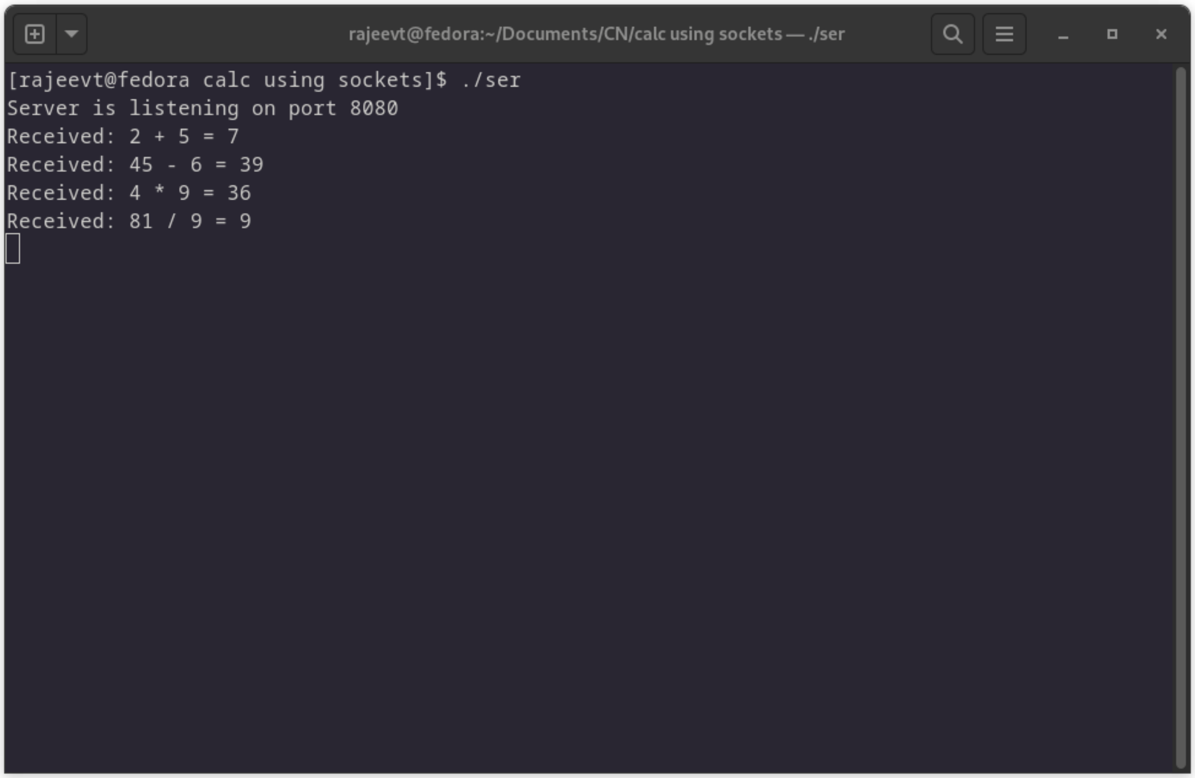
Client Side:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

#define PORT 8080

int main() {
    int client_socket;
    struct sockaddr_in server_address;

    if ((client_socket = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
        perror("socket failed");
        exit(EXIT_FAILURE);
    }

    server_address.sin_family = AF_INET;
    server_address.sin_port = htons(PORT);

    if (inet_pton(AF_INET, "127.0.0.1", &server_address.sin_addr) <= 0) {
        perror("Invalid address / Address not supported");
        exit(EXIT_FAILURE);
    }

    if (connect(client_socket, (struct sockaddr *)&server_address,
sizeof(server_address)) < 0) {
        perror("connection failed");
        exit(EXIT_FAILURE);
    }

```

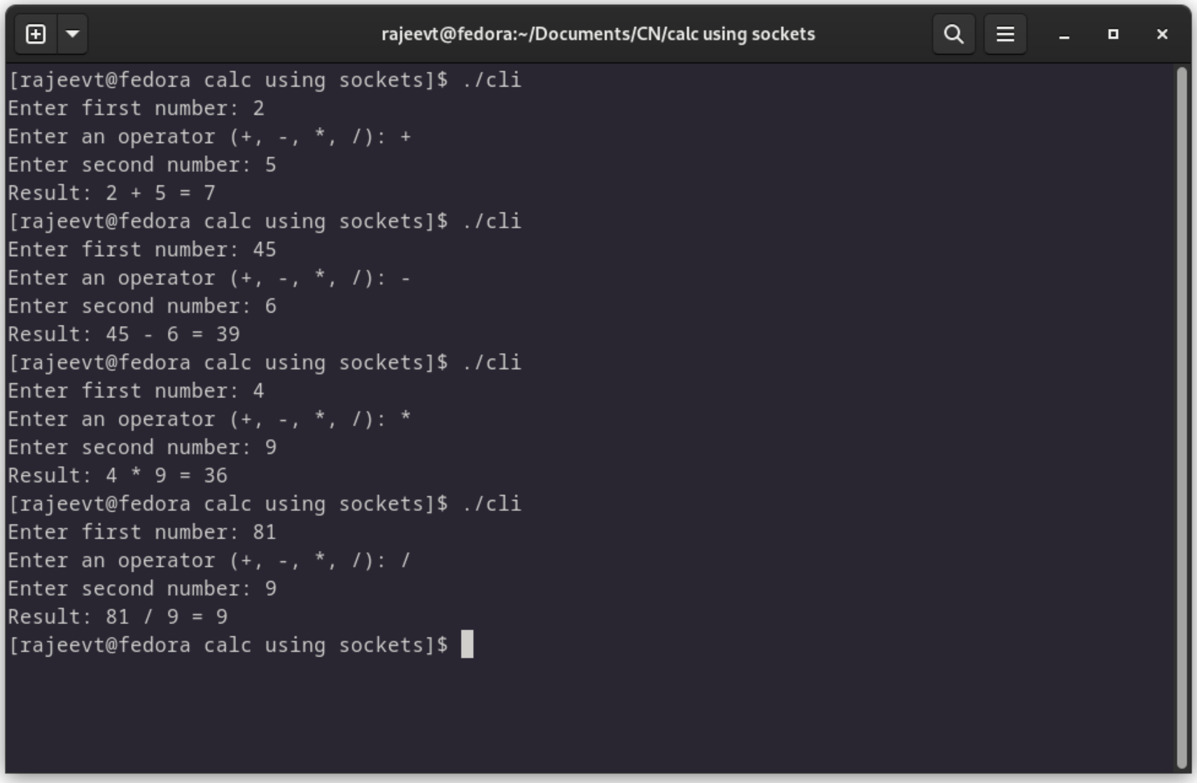


```c
    int num1, num2, result;
    char operator;

    printf("Enter first number: ");
    scanf("%d", &num1);

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

    printf("Enter second number: ");
    scanf("%d", &num2);

    write(client_socket, &num1, sizeof(int));
    write(client_socket, &num2, sizeof(int));
    write(client_socket, &operator, sizeof(char));

    read(client_socket, &result, sizeof(int));

    if (result == -1) {
        printf("Error: Invalid operator or division by zero.\n");
    } else {
        printf("Result: %d %c %d = %d\n", num1, operator, num2, result);
    }

    close(client_socket);

    return 0;
}
```

Output:



```
[rajeevt@fedora calc using sockets]$ ./ser
Server is listening on port 8080
Received: 2 + 5 = 7
Received: 45 - 6 = 39
Received: 4 * 9 = 36
Received: 81 / 9 = 9
```



```
[rajeevt@fedora calc using sockets]$ ./cli
Enter first number: 2
Enter an operator (+, -, *, /): +
Enter second number: 5
Result: 2 + 5 = 7
[rajeevt@fedora calc using sockets]$ ./cli
Enter first number: 45
Enter an operator (+, -, *, /): -
Enter second number: 6
Result: 45 - 6 = 39
[rajeevt@fedora calc using sockets]$ ./cli
Enter first number: 4
Enter an operator (+, -, *, /): *
Enter second number: 9
Result: 4 * 9 = 36
[rajeevt@fedora calc using sockets]$ ./cli
Enter first number: 81
Enter an operator (+, -, *, /): /
Enter second number: 9
Result: 81 / 9 = 9
[rajeevt@fedora calc using sockets]$
```