Experiment 7.

AIM: Write a C Program to implement Sliding window protocol for Selective repeat.

PROGRAM:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define WINDOW_SIZE 4 // Example window size

void sender_sr(int total_frames) {
    int next_frame_to_send = 0;
    int base = 0; // The oldest unacknowledged frame
    int sent_frames[total_frames]; // To keep track of sent frames

    // Initialize sent_frames array
    for (int i = 0; i < total_frames; i++) {
        sent_frames[i] = 0; // 0 means not yet sent/acked
    }
    printf("Sender: Starting Selective Repeat transmission...\n");

    while (base < total_frames) {
        // Send frames within the window
        for (int i = 0; i < WINDOW_SIZE && (next_frame_to_send < total_frames); i++) {
            if (sent_frames[next_frame_to_send] == 0) { // Only send if not already sent/acked
                printf("Sender: Sending Frame %d\n", next_frame_to_send);
                sent_frames[next_frame_to_send] = 1; // Mark as sent
            }
            next_frame_to_send++;
        }

        // Simulate acknowledgments and potential retransmissions
        // For simplicity, we'll randomly simulate ACKs for frames within the window
        // In a real scenario, this involves timers and actual network communication
        for (int i = base; i < next_frame_to_send && i < total_frames; i++) {
            if (sent_frames[i] == 1) { // If frame was sent and not yet acknowledged
                if (rand() % 3 != 0) { // Simulate successful ACK
                    printf("Receiver: Acknowledgment received for Frame %d\n", i);
                    sent_frames[i] = 2; // Mark as acknowledged
                } else { // Simulate lost ACK or damaged frame, requiring retransmission
                    printf("Sender: Timeout/NAK for Frame %d. Retransmitting...\n", i);
                    sent_frames[i] = 0; // Mark for retransmission
                }
            }
        }
        // Slide the window forward if frames at the base are acknowledged
        while (base < total_frames && sent_frames[base] == 2) {
            base++;
        }
        printf("\n");
        // Reset next_frame_to_send to base to re-evaluate sending opportunities
        next_frame_to_send = base;
    }
    printf("Sender: All frames transmitted successfully using Selective Repeat.\n");
}

int main() {
    srand(time(NULL)); // Seed for random number generation
    int total_frames;
    printf("Enter the total number of frames to send: ");
    scanf("%d", &total_frames);
    printf("\n--- Go-Back-N Protocol ---\n");
    sender_sr(total_frames);
    printf("\n--- Selective Repeat Protocol ---\n");
    sender_sr(total_frames);
    return 0;
}
```

OUTPUT