21BPS1191

SPRIHA ANVI

COMPUTER NETWORKS LAB ASSIGNMENT
CHECKSUM ERROR DETECTION

CODE:

SERVER:

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

#define MAX_BUFFER_SIZE 1024

unsigned char calculateChecksum(char *data, int length) {
    unsigned char checksum = 0;

    for (int i = 0; i < length; i++) {
        checksum ^= data[i];
    }
```

```
    return checksum;
}

int main() {
    int sockfd, newSockfd;
    struct sockaddr_in serverAddr, clientAddr;
    socklen_t addrSize;
    char buffer[MAX_BUFFER_SIZE];
    unsigned char receivedData[4];
    unsigned char receivedChecksum, calculatedChecksum;

    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    if (sockfd < 0) {
        perror("Error in socket");
        exit(1);
    }
    serverAddr.sin_family = AF_INET;
    serverAddr.sin_port = htons(12345); // Server port
    serverAddr.sin_addr.s_addr = INADDR_ANY;
    if (bind(sockfd, (struct sockaddr *)&serverAddr,
sizeof(serverAddr)) < 0) {
        perror("Error in bind");
        exit(1);
    }
```

```
    if (listen(sockfd, 10) < 0) {
        perror("Error in listen");
        exit(1);
    }

    printf("Server listening on port 12345...\n");

    addrSize = sizeof(clientAddr);
    newSockfd = accept(sockfd, (struct sockaddr *)&clientAddr,
&addrSize);
    if (newSockfd < 0) {
        perror("Error in accept");
        exit(1);
    }
    read(newSockfd, receivedData, sizeof(receivedData));
    printf("Received data from client.\n");
    printf("Data received from client: %d %d %d %d\n",
receivedData[0], receivedData[1], receivedData[2], receivedData[3]);

    read(newSockfd, &receivedChecksum,
sizeof(receivedChecksum));

    calculatedChecksum = calculateChecksum((char *)receivedData,
sizeof(receivedData));
    if (receivedChecksum == calculatedChecksum) {
        printf("Checksum matched. No error detected.\n");
```

```
    } else {
        printf("Checksum mismatch. Error detected in data.\n");
    }

    close(newSockfd);
    close(sockfd);

    return 0;
}
```

CLIENT:

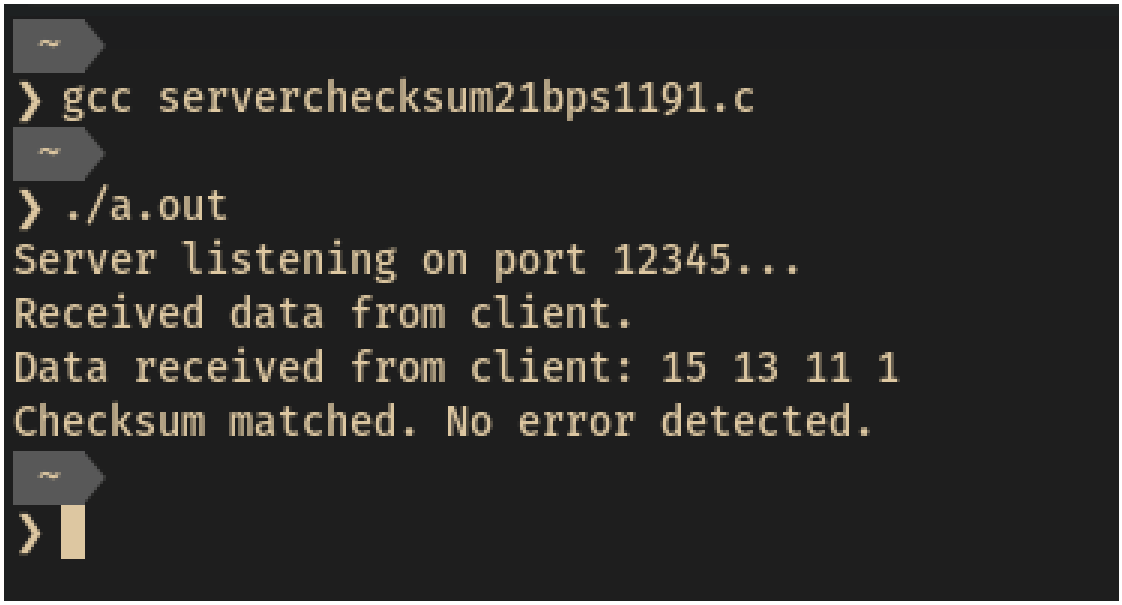
```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>

#define MAX_BUFFER_SIZE 1024
unsigned char calculateChecksum(char *data, int length) {
    unsigned char checksum = 0;

    for (int i = 0; i < length; i++) {
        checksum ^= data[i];
    }
```

```
    return checksum;
}

int main() {
    int sockfd;
    struct sockaddr_in serverAddr;

    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    if (sockfd < 0) {
        perror("Error in socket");
        exit(1);
    }
    serverAddr.sin_family = AF_INET;
    serverAddr.sin_port = htons(12345); // Server port
    serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1"); // Server IP
address
    if (connect(sockfd, (struct sockaddr *)&serverAddr,
sizeof(serverAddr)) < 0) {
        perror("Error in connect");
        exit(1);
    }
    unsigned char data[4] = { 0b1111, 0b1101, 0b1011, 0b0001 };
    printf("Data to be sent to the server: 1111 1101 1011 0001\n");
    unsigned char checksum = calculateChecksum((char *)data,
sizeof(data));
    printf("Checksum: %d\n", checksum);
```

```
    write(sockfd, data, sizeof(data));
    write(sockfd, &checksum, sizeof(checksum));

    printf("Data sent to the server.\n");

    close(sockfd);

    return 0;
}
```

OUTPUT:

SERVER:



CLIENT:

```
 1 ❯ gcc clientchecksum21bps1191.c
 ~    receivedData[2], receivedData[3]);
❯ ./a.out
Data to be sent to the server: 1111 1101 1011 0001
Checksum: 8
Data sent to the server.
 ~
❯
```