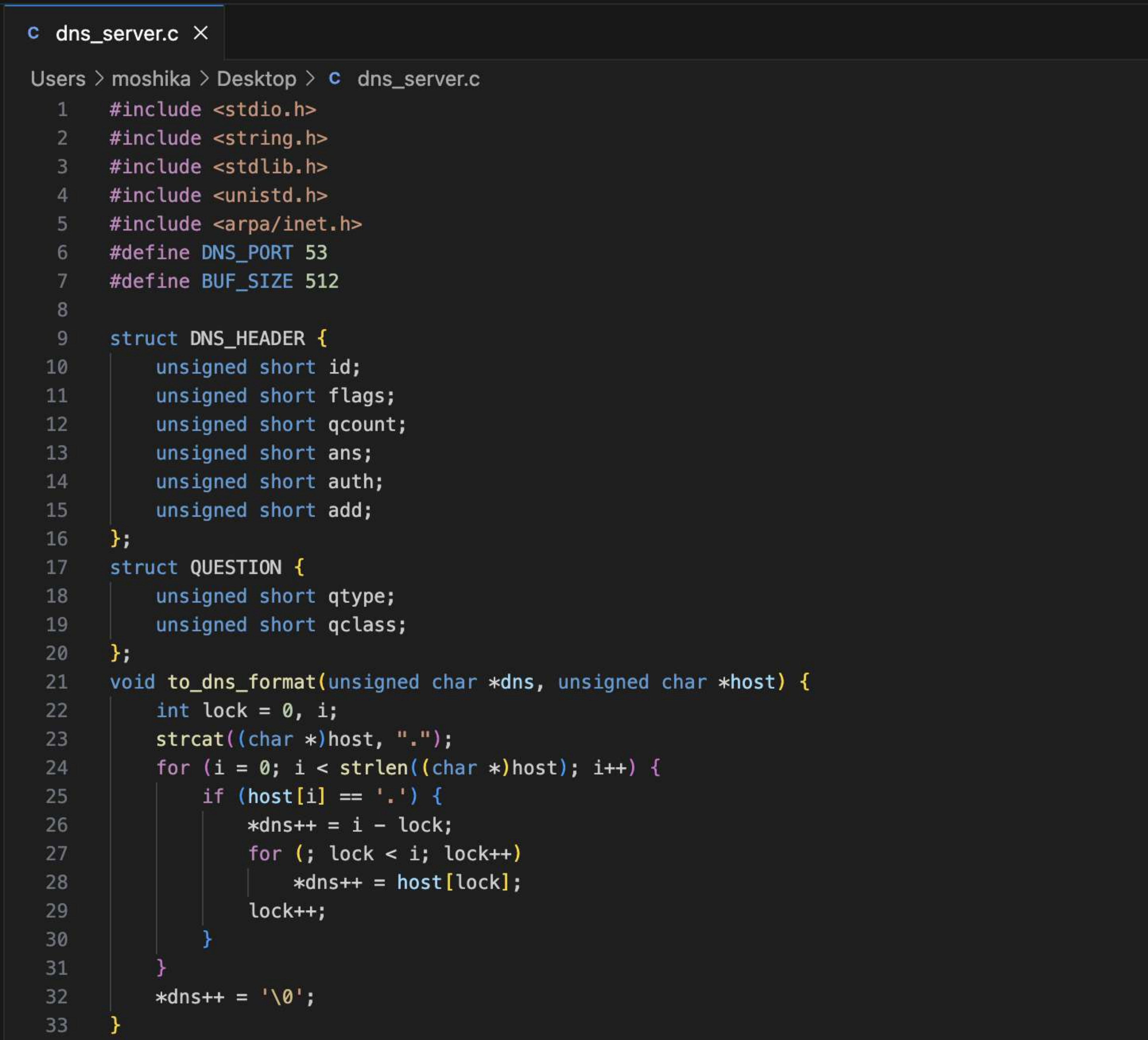
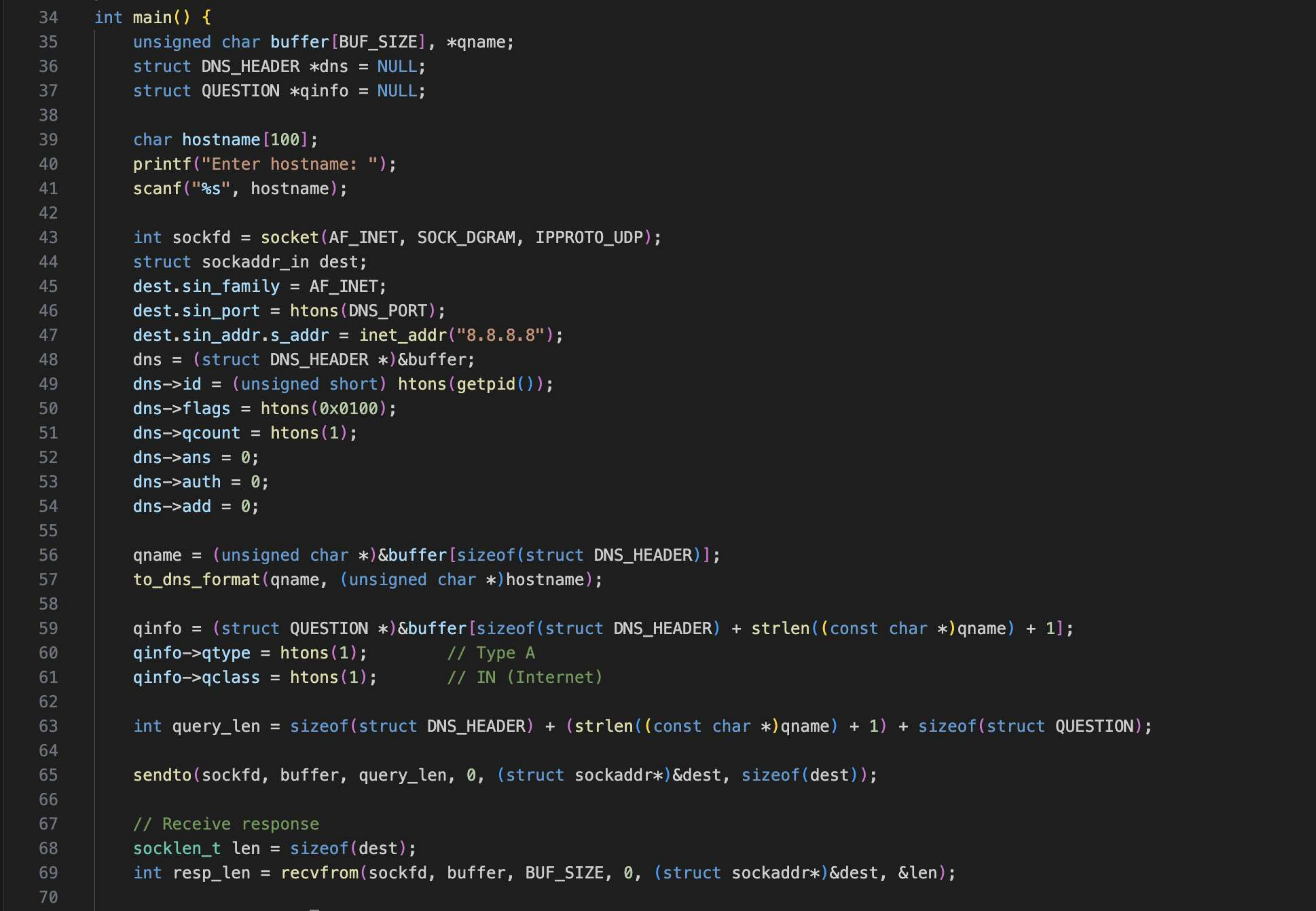
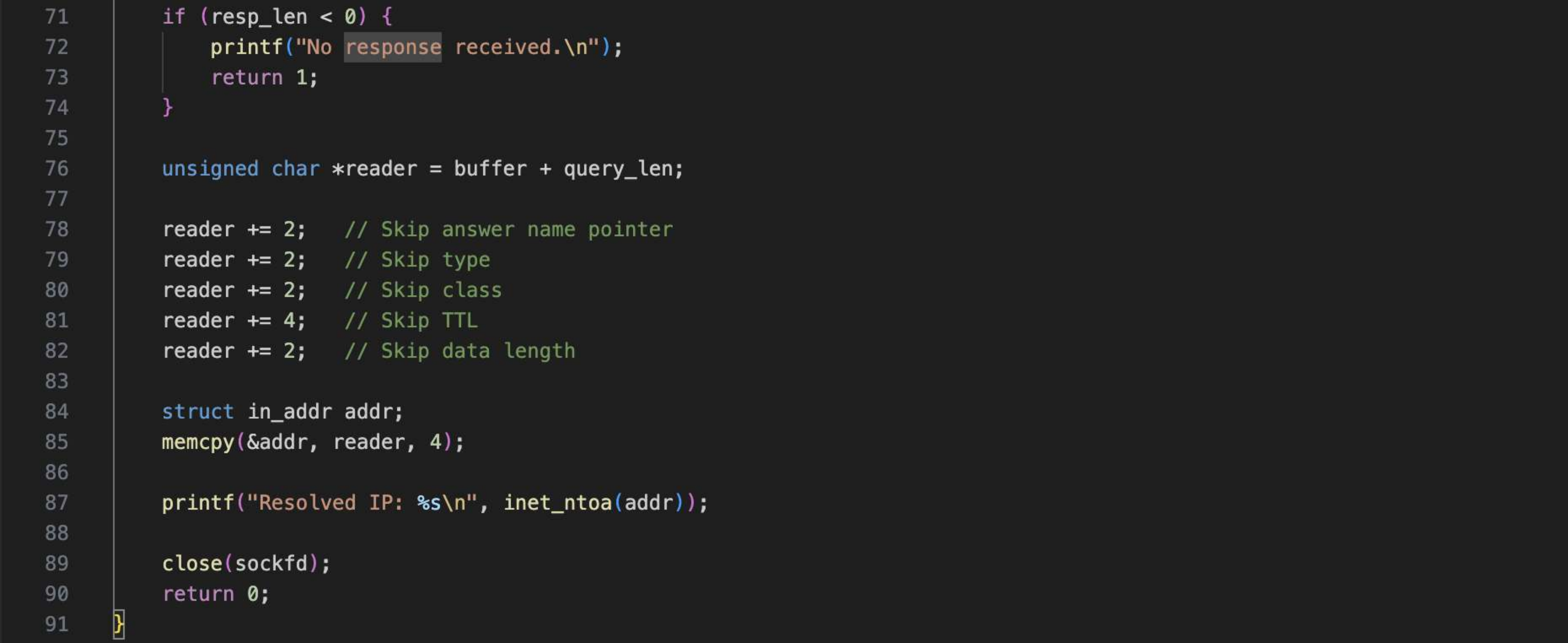


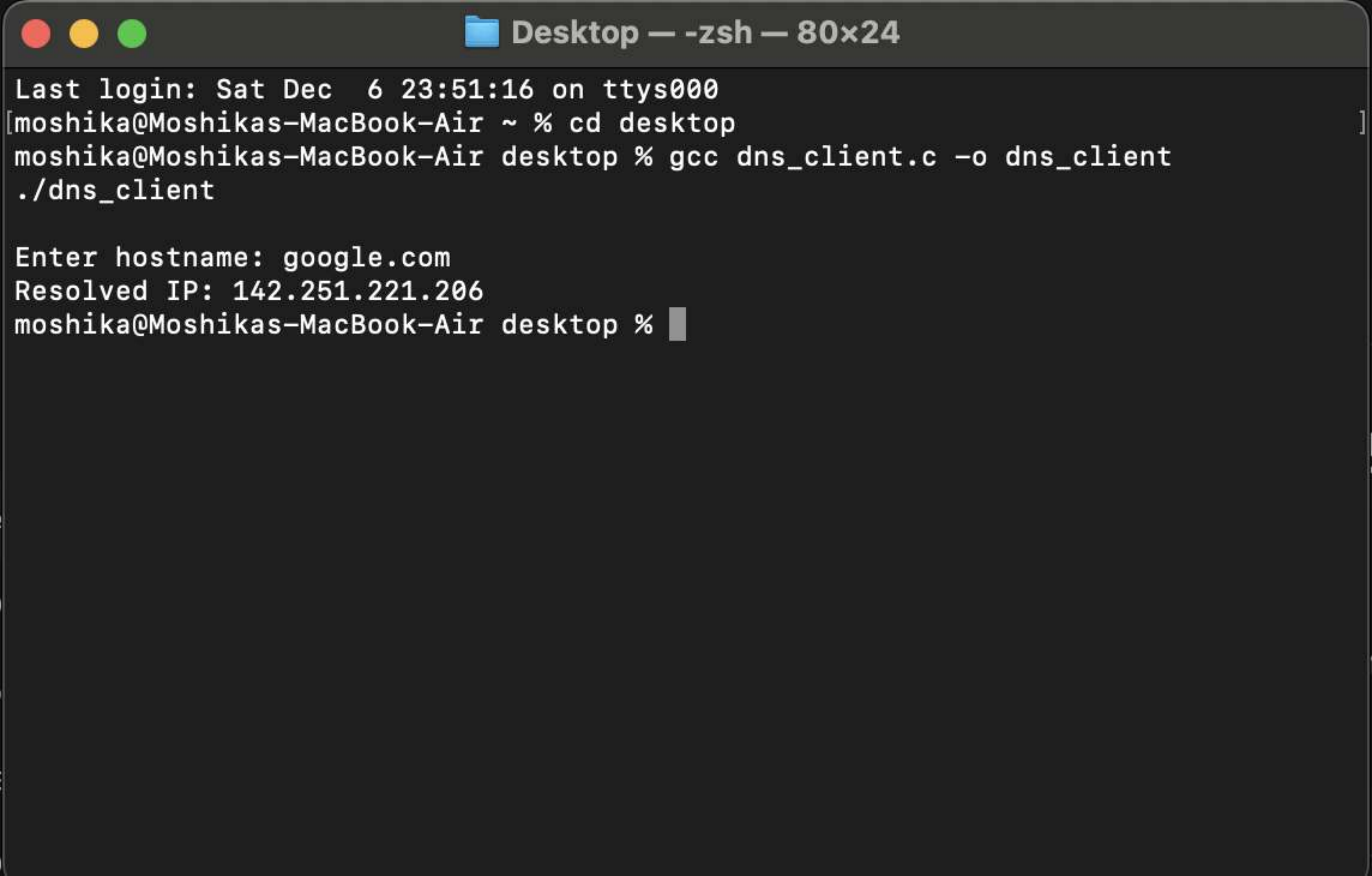
```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>
#include <arpa/inet.h>
#define DNS_PORT 53
#define BUF_SIZE 512

struct DNS_HEADER {
    unsigned short id;
    unsigned short flags;
    unsigned short qcount;
    unsigned short ans;
    unsigned short auth;
    unsigned short add;
};
struct QUESTION {
    unsigned short qtype;
    unsigned short qclass;
};
void to_dns_format(unsigned char *dns, unsigned char *host) {
    int lock = 0, i;
    strcat((char *)host, ".");
    for (i = 0; i < strlen((char *)host); i++) {
        if (host[i] == '.') {
            *dns++ = i - lock;
            for (; lock < i; lock++)
                *dns++ = host[lock];
            lock++;
        }
    }
    *dns++ = '\0';
}
```



```c
int main() {
    unsigned char buffer[BUF_SIZE], *qname;
    struct DNS_HEADER *dns = NULL;
    struct QUESTION *qinfo = NULL;

    char hostname[100];
    printf("Enter hostname: ");
    scanf("%s", hostname);

    int sockfd = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP);
    struct sockaddr_in dest;
    dest.sin_family = AF_INET;
    dest.sin_port = htons(DNS_PORT);
    dest.sin_addr.s_addr = inet_addr("8.8.8.8");
    dns = (struct DNS_HEADER *)&buffer;
    dns->id = (unsigned short) htons(getpid());
    dns->flags = htons(0x0100);
    dns->qcount = htons(1);
    dns->ans = 0;
    dns->auth = 0;
    dns->add = 0;

    qname = (unsigned char *)&buffer[sizeof(struct DNS_HEADER)];
    to_dns_format(qname, (unsigned char *)hostname);

    qinfo = (struct QUESTION *)&buffer[sizeof(struct DNS_HEADER) + strlen((const char *)qname) + 1];
    qinfo->qtype = htons(1);        // Type A
    qinfo->qclass = htons(1);       // IN (Internet)

    int query_len = sizeof(struct DNS_HEADER) + (strlen((const char *)qname) + 1) + sizeof(struct QUESTION);

    sendto(sockfd, buffer, query_len, 0, (struct sockaddr*)&dest, sizeof(dest));

    // Receive response
    socklen_t len = sizeof(dest);
    int resp_len = recvfrom(sockfd, buffer, BUF_SIZE, 0, (struct sockaddr*)&dest, &len);

```



```c
    if (resp_len < 0) {
        printf("No response received.\n");
        return 1;
    }

    unsigned char *reader = buffer + query_len;

    reader += 2;   // Skip answer name pointer
    reader += 2;   // Skip type
    reader += 2;   // Skip class
    reader += 4;   // Skip TTL
    reader += 2;   // Skip data length

    struct in_addr addr;
    memcpy(&addr, reader, 4);

    printf("Resolved IP: %s\n", inet_ntoa(addr));

    close(sockfd);
    return 0;
}
```



```
Last login: Sat Dec  6 23:51:16 on ttys000
moshika@Moshikas-MacBook-Air ~ % cd desktop
moshika@Moshikas-MacBook-Air desktop % gcc dns_client.c -o dns_client
./dns_client

Enter hostname: google.com
Resolved IP: 142.251.221.206
moshika@Moshikas-MacBook-Air desktop % 
```