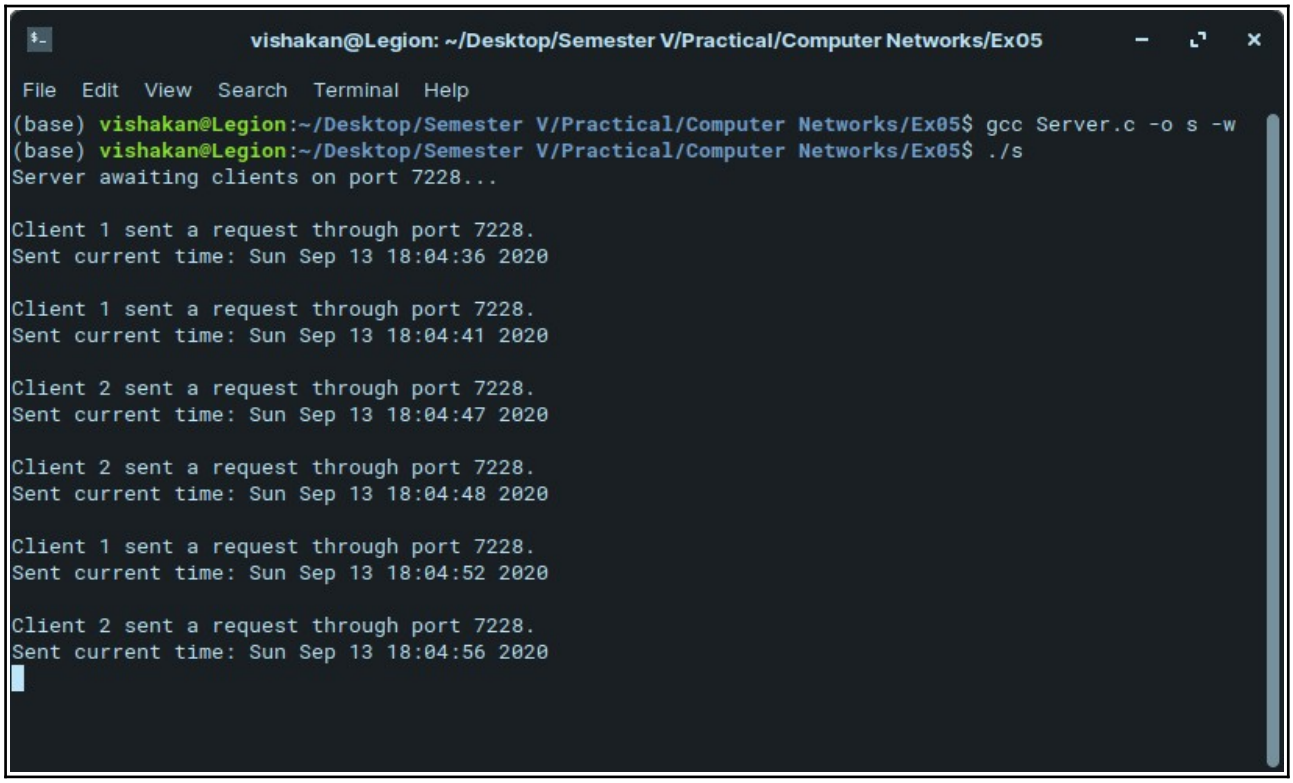
EX5 – DAY TIME SERVER USING UDP

- S. Vishakan CSE – C 18 5001 196

Server Program:

```
#include <stdio.h>
#include <string.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include <sys/socket.h>
#include <netinet/in.h>
#define PORT 7228

int main(int argc, char **argv){
        time_t cur_time;
        struct sockaddr_in serv_addr, cli_addr;
        int sockfd, n, addrlen;
        sockfd = socket(AF_INET, SOCK_DGRAM, 0);

        if(sockfd < 0){
                perror("Error in opening socket.\n");
                exit(1);
        }

        bzero(&serv_addr, sizeof(serv_addr));
        serv_addr.sin_family = AF_INET;
        serv_addr.sin_addr.s_addr = INADDR_ANY;
        serv_addr.sin_port = htons(PORT);

        if(bind(sockfd, (struct sockaddr*)&serv_addr, sizeof(serv_addr)) < 0){
                //Binding the socket to the port with serv_addr
                perror("Bind error occurred.\n");
                exit(1);
        }

        printf("Server awaiting clients on port %d...\n", PORT);
        addrlen = sizeof(cli_addr);
        while(1){ //server is always up
                recvfrom(sockfd, &n, sizeof(n), 0, (struct sockaddr*)&cli_addr, &addrlen);
                cur_time = time(NULL);
                printf("\nClient %d sent a request through port %d.\n", n, PORT);
                sendto(sockfd, &cur_time, sizeof(cur_time), 0, (struct sockaddr*)&cli_addr,
                addrlen);
                printf("Sent current time: %s", ctime(&cur_time));
        }
return 0;
}
```

Output:


vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex05
File Edit View Search Terminal Help
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ gcc Server.c -o s -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ ./s
Server awaiting clients on port 7228...
Client 1 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:36 2020
Client 1 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:41 2020
Client 2 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:47 2020
Client 2 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:48 2020
Client 1 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:52 2020
Client 2 sent a request through port 7228.
Sent current time: Sun Sep 13 18:04:56 2020

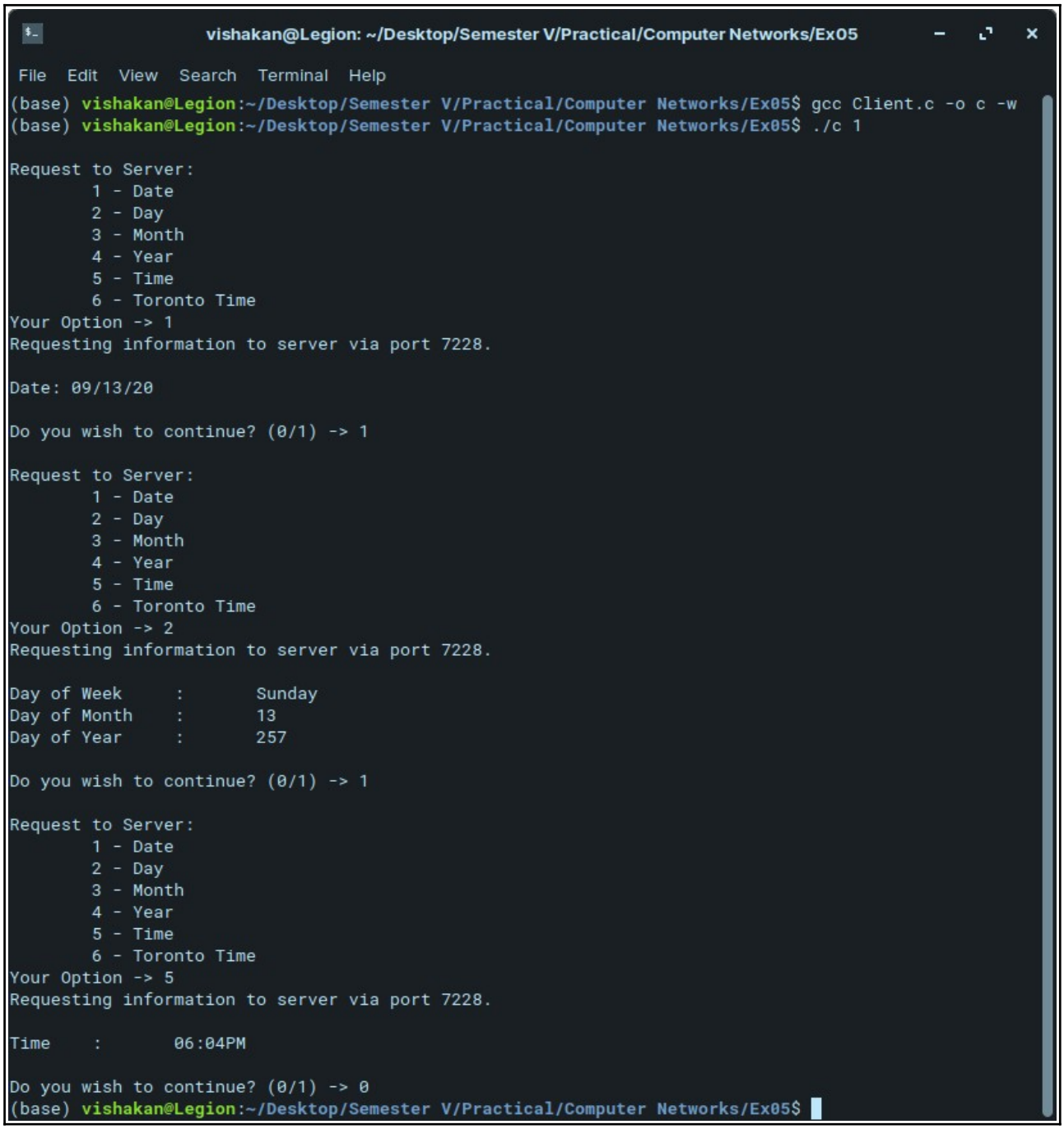
Client Program:

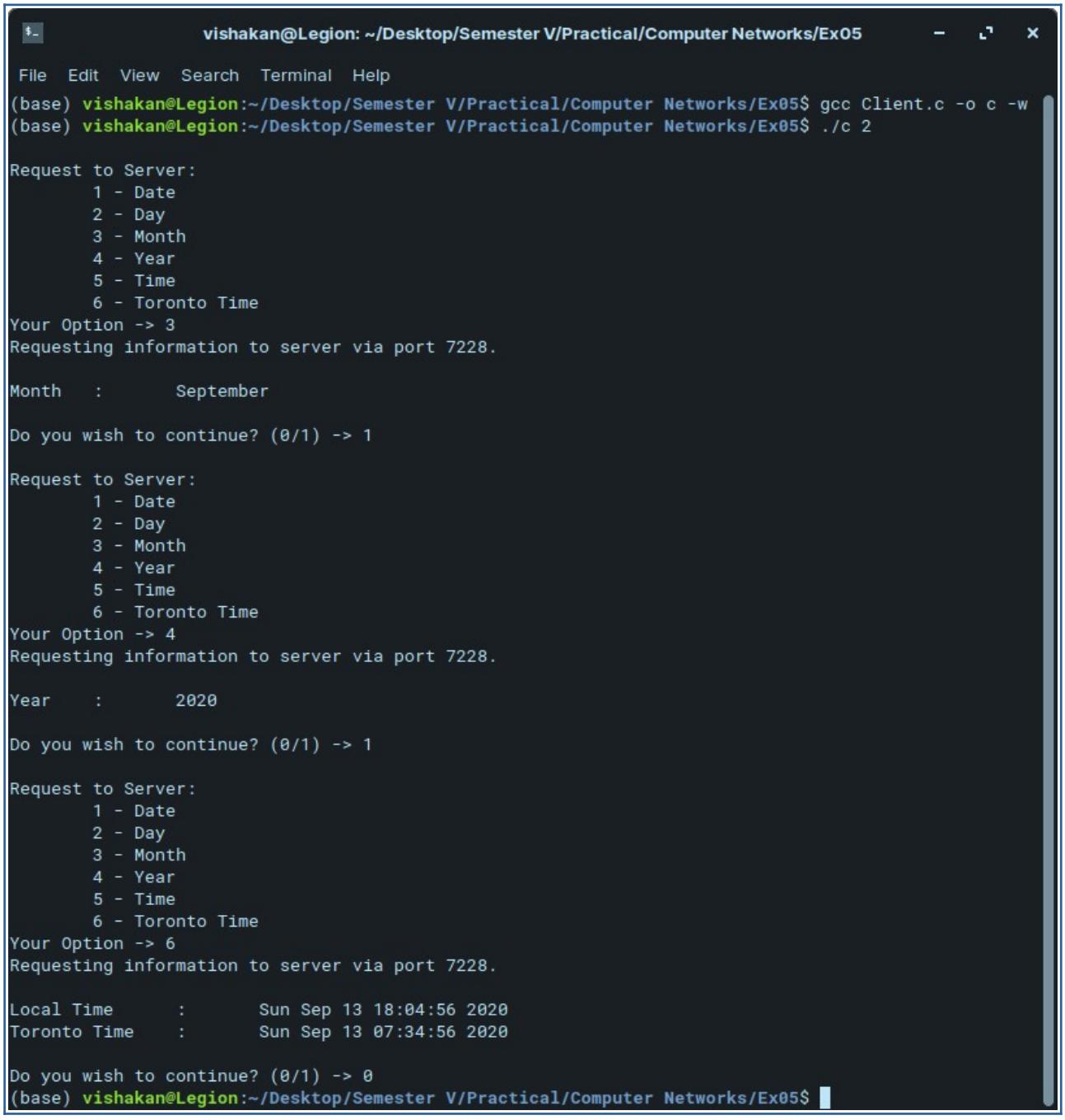
```
#include <stdio.h>
#include <string.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include <sys/socket.h>
#include <netinet/in.h>

#define PORT 7228

void printOption(time_t cur_time, int opt);

int main(int argc, char **argv){
        time_t cur_time;
        struct sockaddr_in serv_addr, cli_addr;
        int sockfd, n, addrlen, opt, cont;
        n = atoi(argv[1]);

        sockfd = socket(AF_INET, SOCK_DGRAM, 0);

        if(sockfd < 0){
                perror("Error in opening socket.\n");
                exit(1);
        }

        bzero(&serv_addr, sizeof(serv_addr));
        serv_addr.sin_family = AF_INET;
        serv_addr.sin_addr.s_addr = INADDR_ANY;
        serv_addr.sin_port = htons(PORT);

        addrlen = sizeof(serv_addr);

        while(1){
                printf("\nRequest to Server:\n\t1 - Date\n\t2 - Day\n\t3 - Month\n\t4 - Year\n\t5
                - Time\n\t6 - Toronto Time\nYour Option -> ");

                scanf("%d", &opt);
                printf("Requesting information to server via port %d.\n\n", PORT);
                sendto(sockfd, &n, sizeof(n), 0, (struct sockaddr*)&serv_addr,
                sizeof(serv_addr));

                recvfrom(sockfd, &cur_time, sizeof(cur_time), 0, (struct
                sockaddr*)&serv_addr, &addrlen);

                printOption(cur_time, opt);
                printf("\nDo you wish to continue? (0/1) -> ");
                scanf("%d", &cont);
```

```
                if(cont == 0){
                        break;
                }
        }

close(sockfd); //close client sockfd once requests are over.

return 0;
}

void printOption(time_t cur_time, int opt){
        struct tm *temp;
        time_t toronto_time;
        char time_buffer[1000];

        temp = localtime(&cur_time);

        switch(opt){
                case 1:
                        strftime(time_buffer, sizeof(time_buffer), "%x", temp);
                        printf("Date: %s\n", time_buffer);
                        break;

                case 2:
                        strftime(time_buffer, sizeof(time_buffer), "%A", temp);
                        printf("Day of Week\t:\t%s\n", time_buffer);
                        strftime(time_buffer, sizeof(time_buffer), "%d", temp);
                        printf("Day of Month\t:\t%s\n", time_buffer);
                        bzero(&time_buffer, sizeof(time_buffer));
                        strftime(time_buffer, sizeof(time_buffer), "%j", temp);
                        printf("Day of Year\t:\t%s\n", time_buffer);
                        break;

                case 3:
                        strftime(time_buffer, sizeof(time_buffer), "%B", temp);
                        printf("Month\t:\t%s\n", time_buffer);
                        break;

                case 4:
                        printf("Year\t:\t%d\n", (temp->tm_year + 1900)); //tm_year stores
                                                                         years elapsed since Unix epoch
                        break;

                case 5:
                        strftime(time_buffer, sizeof(time_buffer), "%I:%M%p", temp);
                        printf("Time\t:\t%s\n", time_buffer);
                        break;
```

```
                case 6:
                        strftime(time_buffer, sizeof(time_buffer), "%c", temp);
                        printf("Local Time\t:\t%s\n", time_buffer);
                        bzero(&time_buffer, sizeof(time_buffer));
                        temp = gmtime(&cur_time); //GMT
                        temp->tm_hour -= 5; //Toronto is -5H ahead of GMT
                        //temp->tm_min -= 30;
                        toronto_time = mktime(temp); //Converting the date & time
                                                        appropriately
                        temp = localtime(&toronto_time);
                        strftime(time_buffer, sizeof(time_buffer), "%c", temp);
                        printf("Toronto Time\t:\t%s\n", time_buffer);
                        break;

                default:
                        printf("\n\tInvalid option.\n");
                        break;
        }

}
```

Output:


vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex05
File Edit View Search Terminal Help
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ gcc Client.c -o c -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ ./c 1
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 1
Requesting information to server via port 7228.
Date: 09/13/20
Do you wish to continue? (0/1) -> 1
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 2
Requesting information to server via port 7228.
Day of Week : Sunday
Day of Month : 13
Day of Year : 257
Do you wish to continue? (0/1) -> 1
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 5
Requesting information to server via port 7228.
Time : 06:04PM
Do you wish to continue? (0/1) -> 0
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$


vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex05
File Edit View Search Terminal Help
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ gcc Client.c -o c -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$ ./c 2
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 3
Requesting information to server via port 7228.
Month : September
Do you wish to continue? (0/1) -> 1
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 4
Requesting information to server via port 7228.
Year : 2020
Do you wish to continue? (0/1) -> 1
Request to Server:
1 - Date
2 - Day
3 - Month
4 - Year
5 - Time
6 - Toronto Time
Your Option -> 6
Requesting information to server via port 7228.
Local Time : Sun Sep 13 18:04:56 2020
Toronto Time : Sun Sep 13 07:34:56 2020
Do you wish to continue? (0/1) -> 0
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex05$