Shri Vile Parle Kelavani Mandal's

INSTITUTE OF TECHNOLOGY

DHULE (M.S.)

DEPARMENT OF COMPUTER ENGINEERING

Subject : Competitive Programming Lab

Name: Mohammed Meraj Mohammed Ashfaque **Roll No. :** 32

Class : TY. Comp. Engg. **Batch : T2** **Division: T**

Expt. No. :09 **Date : 24/03/2025**

Title : Longest Nap Problem

Remark

Signature

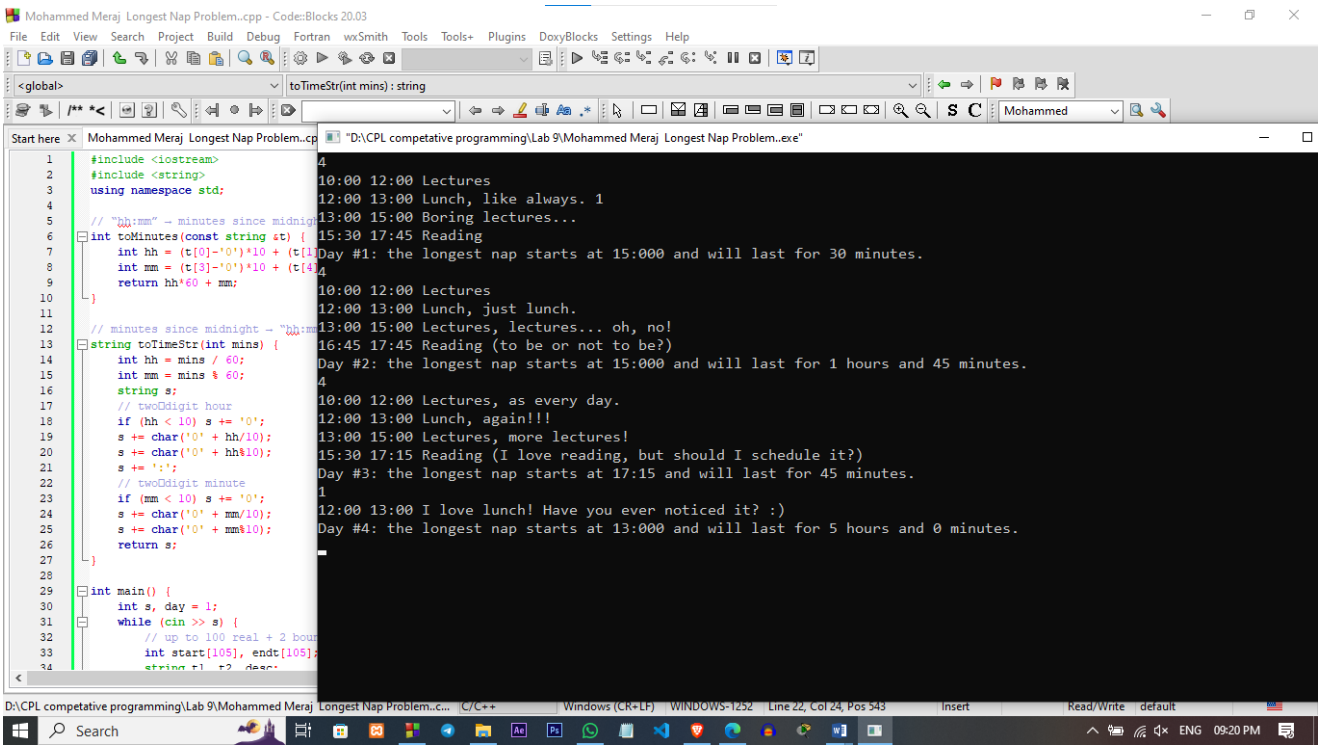
Language: C++

```
// Longest Nap Problem by Meraj 32 T2
#include <iostream>
#include <string>
using namespace std;

// “hh:mm” → minutes since midnight
int toMinutes(const string &t) {
    int hh = (t[0]-'0')*10 + (t[1]-'0');
    int mm = (t[3]-'0')*10 + (t[4]-'0');
    return hh*60 + mm;
}

// minutes since midnight → “hh:mm”
string toTimeStr(int mins) {
    int hh = mins / 60;
    int mm = mins % 60;
    string s;
    // two-digit hour
    if (hh < 10) s += '0';
    s += char('0' + hh/10);
    s += char('0' + hh%10);
    s += ':';
    // two-digit minute
```

```
    if (mm < 10) s += '0';
    s += char('0' + mm/10);
    s += char('0' + mm%10);
    return s;
}

int main() {
    int s, day = 1;
    while (cin >> s) {
        // up to 100 real + 2 boundary
        int start[105], endt[105];
        string t1, t2, desc;

        // read appointments
        for (int i = 1; i <= s; ++i) {
            cin >> t1 >> t2;
            getline(cin, desc);  // ignore rest of line
            start[i] = toMinutes(t1);
            endt[i]  = toMinutes(t2);
        }
        // day bounds
        start[0] = endt[0] = toMinutes("10:00");
        start[s+1] = endt[s+1] = toMinutes("18:00");
        int n = s + 2;

        // selection sort by start time
        for (int i = 0; i < n-1; ++i) {
            int minIdx = i;
            for (int j = i+1; j < n; ++j) {
                if (start[j] < start[minIdx]) {
                    minIdx = j;
                }
            }
            // swap starts
            int tmp = start[i]; start[i] = start[minIdx]; start[minIdx] = tmp;
            // swap ends
            tmp = endt[i];      endt[i]   = endt[minIdx];   endt[minIdx]   = tmp;
        }

        // find longest free interval
        int bestLen = -1, bestStart = 0;
        for (int i = 0; i < n-1; ++i) {
            int freeTime = start[i+1] - endt[i];
```

```
            if (freeTime > bestLen) {
                bestLen = freeTime;
                bestStart = endt[i];
            }
        }

        // split into hours/minutes
        int H = bestLen / 60;
        int M = bestLen % 60;

        // output
        cout << "Day #" << day++
             << ": the longest nap starts at "
             << toTimeStr(bestStart)
             << " and will last for ";
        if (bestLen < 60) {
            cout << M << " minutes.";
        } else {
            cout << H << " hours and " << M << " minutes.";
        }
        cout << '\n';
    }
    return 0;
}
```

Output :


Mohammed Meraj Longest Nap Problem..cpp - Code::Blocks 20.03
"D:\CPL competative programming\Lab 9\Mohammed Meraj Longest Nap Problem..exe"
4
10:00 12:00 Lectures
12:00 13:00 Lunch, like always. 1
13:00 15:00 Boring lectures...
15:30 17:45 Reading
Day #1: the longest nap starts at 15:000 and will last for 30 minutes.
4
10:00 12:00 Lectures
12:00 13:00 Lunch, just lunch.
13:00 15:00 Lectures, lectures... oh, no!
16:45 17:45 Reading (to be or not to be?)
Day #2: the longest nap starts at 15:000 and will last for 1 hours and 45 minutes.
4
10:00 12:00 Lectures, as every day.
12:00 13:00 Lunch, again!!!
13:00 15:00 Lectures, more lectures!
15:30 17:15 Reading (I love reading, but should I schedule it?)
Day #3: the longest nap starts at 17:15 and will last for 45 minutes.
1
12:00 13:00 I love lunch! Have you ever noticed it? :)
Day #4: the longest nap starts at 13:000 and will last for 5 hours and 0 minutes.