# C++ Fundamentals: Judge Assignment 1 (JA1)

The following tasks should be submitted to the SoftUni Judge system, which will be open starting Saturday, 26 May 2018, 10:00 (in the morning) and will close on Sunday, 10 June 2018, 23:59. Submit your solutions here: <https://judge.softuni.bg/Contests/Compete/Index/1037> .

After the system closes, you will be able to “Practice” on the tasks – however the “Practice” results are NOT considered in the homework evaluation.

For this assignment, the code for each task should be a single C++ file, the contents of which you copy-paste into the Judge system.

Please be mindful of the strict input and output requirements for each task, as well as any additional requirements on running time, used memory, etc., as the tasks are evaluated automatically and not following the requirements strictly may result in your program’s output being evaluated as incorrect, even if the program’s logic is mostly correct.

You can use C++03 and C++11 features in your code.

Unless explicitly stated, any integer input fits into int and any floating-point input can be stored in double.

NOTE: the tasks here are NOT ordered by difficulty level.

## Task 3 – Bus (JA1-Task-3-Bus)

Captain Grant needs your help. He’s currently on leave, but needs to get back to his ship soon. To do that, he needs to catch a bus to the train station, and from there take a train to the naval base. But captain Grant hates waiting – he has a certain train he has to catch, but can pick from several busses, and he wants to pick a bus which arrives as close to the train departure as possible.

The transport company, which operates the busses to the station and the trains at the station, has a list of bus arrival times at the station, as well as information on the train departure time. Of course, since the company works with the military, the arrival times and the train departure time are in military time format – 4-digit numbers, the first two digits represent the hours (00 to 23), the next two digits represent the minutes (00 to 59). For example, two o’clock in the morning is 0200, twenty minutes past four in the afternoon is 1620, two minutes to midnight (*the time, not the Iron Maiden song*) is 2358, etc.

Write a program which, given a list of bus arrival times and a train departure time, in military time format, finds the minimum amount of time – in minutes – between a bus arrival and the train departure (i.e. the time Grant would have to wait if he picks the “best” bus) and prints the position of the bus in the bus arrival times list.

Note that 0 waits is possible, but negative wait times aren’t possible.

*Hint: you can convert the military time format numbers into minutes (minutes elapsed since midnight) before calculating the time between an arrival time and the train time*

### Input

The first line of the standard input will contain the number N – the number of bus arrival times.

The first line of the standard input will contain a sequence of bus arrival times, in military time format, separated by single spaces.

The second line of the standard input will contain the train departure time, in military time format.

### Output

A single line containing a single non-negative integer – the number/position of the bus in the input sequence of bus arrival times, for which the wait time is minimal.

### Restrictions

N will be at least 1 and at most 20.

The input data will be such that there will always be a valid (non-negative) minimum wait time. There will always be a bus that arrives before the train leaves.

The total running time of your program should be no more than 0.1s

The total memory allowed for use by your program is 5MB

### Example I/O

|  |  |  |
| --- | --- | --- |
| Example Input | Expected Output | Explanation |
| 4  2013 0130 0004 0012  2122 | 1 | The best bus is the one arriving at 2013 (20:13) – i.e. the 1st bus in the list (NOTE: the answer is the position of the bus in the input, not in their chronological order) |
| 3  1205 1241 1708  1241 | 2 | The train leaves at 1241 and the 2nd bus arrives then (0 minutes wait, so it is the best option) |