

题目描述：

运维工程师采集到某产品现网运行一天产生的日志 N 条，现需根据日志时间按时间先后顺序对日志进行排序。
日志时间格式为：

$H:M:S.N$

H 表示小时($0-23$)， M 表示分钟($0-59$)， S 表示秒($0-59$)， N 表示毫秒($0-999$)

时间可能并没有补齐，也就是说: $01:01:01.001$ ，也可能表示为 $1:1:1.1$

输入描述：

第一行输入一个整数 N ，表示日志条数， $1 \leq N \leq 100000$

接下来 N 行输入 N 个时间

输出描述：

按时间升序排序之后的时间
如果有两个时间表示的时间相同，则保持输入顺序

补充说明：

示例 1

输入：

2

01:41:8.9

1:1:09.211

输出：

1:1:09.211

01:41:8.9

说明：

示例 2

输入:

3

23:41:08.023

1:1:09.211

08:01:22.0

输出:

1:1:09.211

08:01:22.0

23:41:08.023

说明:

示例 3

输入:

2

22:41:08.023

22:41:08.23

输出:

22:41:08.023

22:41:08.23

说明:

两个时间表示的时间相同，保持输入顺序

```
const rl = require("readline").createInterface({ input: process.stdin });
```

```
var iter = rl[Symbol.asyncIterator]();
```

```
const readline = async () => (await iter.next()).value;
```

```
void async function () {
```

```
    // Write your code here
```

```
    const n = parseInt(await readline())
```

```
    const arr = []
```

```
    const temp = [3600 * 1000, 60 * 1000, 1000, 1]
```

```
    for (let i = 0; i < n; i++) {
```

```
        const line = await readline()
```

```
        const tokens = line.split(/[:.]/).map(Number)
```

```
const val = tokens.reduce((acc, item, index) => {
  acc += item * temp[index]
  return acc
}, 0)
arr.push({
  line,
  val,
})
}
arr.sort((a, b) => {
  return a.val - b.val
})
console.log(arr.map(item => item.line).join("\n"))
}()
```