

响应报文时间

题目描述: IGMP 协议中, 有一个字段称作最大响应时间(Max Response Time), HOST收到查询报文, 解析出 MaxResponseTime 字段后, 需要在 (0, MaxResponseTime](s) 时间内选取随机时间回应一个响应报文, 如果在随机时间内收到一个新的查询报文, 则会根据两者时间的大小, 选取小的一方刷新回应时间。

最大响应时间有如下计算方式:

当 Max Resp Code < 128, Max Resp Time = Max Resp Code;

当 Max Resp Code >= 128,

0 1 2 3 4 5 6 7

+--+--+--+--+

|1| exp | mant |

+--+--+--+--+

Max Resp Time = (mant | 0x10) << (exp + 3);

注: exp最大响应时间的高5~7位; mant 为最大响应时间的低4位。

其中 接收到的MaxRespCode 最大值为 255, 以上出现所有字段均为无符号数。

现在我们认为 HOST 收到查询报文时, 选取的随机时间必定为最大值。现给出 HOST 收到查询报文个数 C, HOST 收到该报文的时间 T, 以及查询报文的最大响应时间字段值 M, 请计算出HOST 发送响应报文的时间。

输入: 第一行为查询报文个数 C, 后续每行分别为 HOST 收到报文时间 T, 及最大响应字段M, 以空格分割。

输出: HOST发送响应报文的时间

输入描述: 第一行为查询报文个数 C, 后续每行分别为 HOST 收到报文时间 T, 及最大响应时间M, 以空格分割。

输出描述: HOST 发送响应报文的时间。

补充说明: 用例确定只会发送一个响应报文, 不存在计时结束后依然收到查询报文的情况。

示例1

输入: 3

0 20

1 10

8 20

输出: 11

说明: 收到3个报文,

第0秒收到第1个报文, 响应时间为20秒, 则要到0+20=20秒响应;

第1秒收到第2个报文, 响应时间为10; 则要到1+10=11秒响应, 与第上面的报文的响应时间比较获得响应时间最小为11秒;

第8秒收到第3个报文, 响应时间为20秒, 则要到8+20=28秒响应; 与第上面的报文的响应时间比较获得响应时间最小为11秒;

最终得到最小响应报文时间为11秒

示例2

输入: 2

0 255

200 60

输出: 260

说明: 收到2个报文,

第0秒收到第1个报文, 响应时间为255秒, 则要到(15 | 0x10) << (7 + 3) = 31744秒响应; (mant = 15, exp = 7)

第200秒收到第2个报文, 响应时间为60; 则要到200+60=260秒响应, 与第上面的报文的响应时间比较获得响应时间最小为260秒;

最终得到最小响应报文时间为260秒

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int n ;
```

```
    cin >> n;
```

```
    int a,b;
```

```
    int ret = 999999999;
```

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

```
        cin >> a >> b;
```

```
        if (b >= 128) {
```

```
            b = ((b & 0xf) | 0x10) << (((b & 0x70) >> 4) + 3);
```

```
        }
```

```
        ret = min(ret, a + b);
```

```
    }
```

```
    cout << ret;
```

```
    return 0;
```

