

每日一题Java方向day37_7月13日

一. 单选

1. 下面关于synflood攻击的说法,错误的是()

- ☐ A 服务端由于连接队列被占满而不能对外服务
- ☐ B 不但能攻击TCP服务,还可以攻击UDP服务
- ☐ C 大量连接处于SYN_RECV状态
- ☐ D 使用硬件防火墙可以一定程度上抵御攻击

正确答案 : B

2. A,B两台机器都正常工作,B机器未监听任何端口.如果A机器向B机器80端口发送SYN包,会收到何种类型的回包?

- ☐ A ACK包
- ☐ B FIN包
- ☐ C 不会收到回包
- ☐ D RST包

正确答案 : D

3. 下列哪个IP地址可以分配给一台计算机?

- ☐ A 256.1.3.4
- ☐ B 197.3.11.0
- ☐ C 199.5.89
- ☐ D 11.15.33.235

正确答案 : D

4. 以下哪个ip不和10.11.12.91/28处于同一个子网

- ☐ A 10.11.12.85/28
- ☐ B 10.11.12.88/28
- ☐ C 10.11.12.94/28
- ☐ D 10.11.12.97/28

正确答案 : D

5. 假设在x86平台上,有一个int型变量,在内存中的内部由低到高分别是:0x12,0x34,0x56,0x78当通过网络发送

该数据时,正确的发送顺序是()

- A 0x12,0x34,0x56,0x78
- B 0x78,0x56,0x34,0x12
- C 0x34,0x12,0x78,0x56
- D 0x56,0x78,0x12,0x34

正确答案 : B

6. 某学校获取到一个B类地址段,要给大家分开子网使用,鉴于现在上网设备急剧增多,管理员给每个网段进行划分的子网掩码设置为255.255.254.0,考虑每个网段需要有网关设备占用一个地址的情况下,每个网段还有多少可用的主机地址()

- A 509
- B 511
- C 512
- D 510

正确答案 : A

7. 下列哪个地址不可能是子网掩码()

- A 255.224.0.0
- B 255.255.240.0
- C 255.255.255.248
- D 255.255.255.250

正确答案 : D

8. 当我们在局域网内使用ping www.nowcoder.com时, 哪种协议没有被使用?

- A ICMP
- B ARP
- C DNS
- D TCP

正确答案 : D

9. 从浏览器打开 http://www.sohu.com , TCP/IP协议族中不会被使用到的协议是()

- A SMTP
- B HTTP
- C TCP

D IP

正确答案：A

10. 某公司申请到一个C类IP地址，但要连接6个的子公司，最大的一个子公司有26台计算机，每个子公司在一个网段中，则子网掩码应设为（ ）

- A 255.255.255.192
- B 255.255.255.128
- C 255.255.255.0
- D 255.255.255.224

正确答案：D

二. 编程

1. 标题：木棒拼图 | 时间限制：1秒 | 内存限制：32768K

有一个由很多木棒构成的集合，每个木棒有对应的长度，请问能否用集合中的这些木棒以某个顺序首尾相连构成一个面积大于 0 的简单多边形且所有木棒都要用上，简单多边形即不会自交的多边形。

初始集合是空的，有两种操作，要么给集合添加一个长度为 L 的木棒，要么删去集合中已经有的某个木棒。每次操作结束后你都需要告知是否能用集合中的这些木棒构成一个简单多边形。

输入描述：

每组测试用例仅包含一组数据，每组数据第一行为一个正整数 n 表示操作的数量($1 \leq n \leq 50000$)，接下来有 n 行，每行第一个整数为操作类型 i ($i \in \{1, 2\}$)，第二个整数为一个长度 L ($1 \leq L \leq 1,000,000,000$)。如果 i=1 代表在集合内插入一个长度为 L 的木棒，如果 i=2 代表删去在集合内的一根长度为 L 的木棒。输入数据保证删除时集合中必定存在长度为 L 的木棒，且任意操作后集合都是非空的。

输出描述：

对于每一次操作结束有一次输出，如果集合内的木棒可以构成简单多边形，输出 "Yes"，否则输出 "No"。

示例1:

输入

```
5
1 1
1 1
1 1
2 1
1 2
```

输出

```
No
No
Yes
No
No
```

正确答案：

```
#include <iostream>
#include <list>
using namespace std;

const char* is_ok(const list<size_t>& stick_set) {
    size_t sum = 0, max_len = 0;
    for (auto& v : stick_set) {
        if (v > max_len) {
            max_len = v;
        }
        sum += v;
    }
    if (sum - max_len <= max_len)
        return "No";
    else
        return "Yes";
}

int main() {
    size_t n = 0;
    cin >> n;

    list<size_t> stick_set;
    while (n--) {
        int opt = 0;
        size_t len = 0;
        cin >> opt >> len;

        if (opt == 1) {
            // 增加一个 len
            stick_set.push_back(len);
        }
        else if (opt == 2) {
            // 删除一个 len
            auto&& it = stick_set.begin();
            for (; it != stick_set.end(); ++it) {
                if (*it == len) {
                    stick_set.erase(it);
                    break;
                }
            }
            if (it == stick_set.end()) cout << "Error\n"; // not found
        }
        else {
            cout << "Error\n"; // Illegal value
        }

        cout << is_ok(stick_set) << endl;
    }
    return 0;
}
```

2. 标题：地下迷宫 | 时间限制：1秒 | 内存限制：32768K

小青蛙有一天不小心落入了一个地下迷宫,小青蛙希望用自己仅剩的体力值P跳出这个地下迷宫。为了让问题简单,假设这是一个 $n*m$ 的格子迷宫,迷宫每个位置为0或者1,0代表这个位置有障碍物,小青蛙达到不了这个位置;1代表小青蛙可以达到的位置。小青蛙初始在(0,0)位置,地下迷宫的出口在(0,m-1)(保证这两个位置都是1,并且保证一定有起点到终点可达的路径),小青蛙在迷宫中水平移动一个单位距离需要消耗1点体力值,向上爬一个单位距离需要消耗3个单位的体力值,向下移动不消耗体力值,当小青蛙的体力值等于0的时候还没有到达出口,小青蛙将无法逃离迷宫。现在需要你帮助小青蛙计算出能否用仅剩的体力值跳出迷宫(即达到(0,m-1)位置)。

输入描述：

输入包括n+1行:

第一行为三个整数n,m($3 \leq m, n \leq 10$),P($1 \leq P \leq 100$)

接下来的n行:

每行m个0或者1,以空格分隔

输出描述 :

如果能逃离迷宫,则输出一行体力消耗最小的路径,输出格式见样例所示;如果不能逃离迷宫,则输出"Can not escape!". 测试数据保证答案唯一

示例1:

输入

4 4 10 1 0 0 1 1 1 0 1 0 1 1 1 0 0 1 1

输出

[0,0],[1,0],[1,1],[2,1],[2,2],[2,3],[1,3],[0,3]

正确答案 :