

全国入门组 CSP-J 初赛模拟试题 (2)

一、单项选择题 (共 15 题, 每题 2 分, 共计 30 分; 每题有且仅有一个正确选项)

1. 以下哪些不是属于国家顶级域名的是 ()。
A..au B..cn C..com D.jp
2. 2 个 10 进制数 1111 和 1010 的异或运算结果的 10 进制表示是 ()。
A.101 B.3 C.1957 D.5
3. 8 位二进制数中去掉符号位, 最大能表示多少字符 ()。
A.127 B.128 C.255 D.256
4. 在写递归函数时, 哪些定义一般不写在递归函数中 ()。
A.int B.float C.double D.数组
5. 一棵完全二叉树, 共有 1234 个节点, 其叶子节点的个数为 ()。
A.615 B.616 C.617 D.210
6. 某公司派赵钱孙李周五人出国学习, 选派条件是:
a.若赵去, 钱也去;
b.李、周两人必有一人去;
c.如周去, 则赵、钱也同去;
d.孙、李二人同去或同不去;
如何选他们出国? ()。
A.孙赵周去 B.赵钱周去 C.李周孙去 D.钱孙去
7. 已知一棵二叉树前序遍历为 ABCDEFGI, 后序遍历为 CEDBIGFA, 则其中序遍历可能为 ()。
A.ABCDEFGI B.CBEDAFIG C.CBDEAGFI D.CBEDAIFG
8. 8 颗子弹, 编号为 1、2、3、4、5、6、7、8, 从编号 1 开始按序嵌入弹夹, 以下有哪个不是正常的打出子弹的次序 ()。
A.12345678 B.87654321 C.32154876 D.32164587
9. 已知循环队列空间为 30, 队头位置编号为 12, 队尾元素下一个空位置编号为 5, 则队伍中元素个数为 ()。
A.22 B.23 C.7 D.8
10. 甲箱中有 200 个螺杆, 其中有 160 个 A 型螺杆; 乙箱中有 240 个螺母, 其中有 180 个 A 型的。现从甲乙两箱中各任取一个, 则能配成 A 型螺栓的概率为多少? ()
A.1/20 B.19/20 C.3/5 D.15/16
11. 今年信息学进复赛的同学有 6 人, 老师将他们排成一圈分发奖品, 请问有几种排法 ()。
A.60 B.120 C.180 D.240
12. 设二维数组 A 的行下标为 0 至 5, 列下标为 1 至 5, F 的每个数据元素均占 2 个字节。在按行存贮的情况下, 已知数据元素 A[3][3] 的第一个字节是 2019, 则 A[4][4] 的第一个字节的地址为 ()。
A.2029 B.2025 C.2027 D.2031
13. 在右图中, 有 () 个顶点出发存在一条路径可以遍历图中得每条边, 而且仅遍历一次。

- A.6 B.2 C.3 D.4




```

2  using namespace std;
3  int main() {
4      int a[1001],i,j,t,n;
5      for(i=0; i<=1000; i++)
6          a[i]=0;
7      scanf("%d",&n);
8      for(i=1; i<=n; i++) {
9          scanf("%d" ,&t);
10         a[t]++;
11     }
12     for(i=1000; i>=0; i--)
13         for(j=1; j<=a[i]; j++)
14             printf("%d ",i);
15     return 0;
16 }

```

判断题

- 1) 输入 10 个数字，输出结果是从小到大。()
- 2) (2 分) 输入的数字中有 2 个 1，则输出时出来第一个 1 是第一个输入。()
- 3) (2 分) 第 13 行的 <= 改为 < 号，且输入数据为 10 2 12 33 34 1 28 7 22 9 0，则无输出。()
- 4) 第 12 行改为 for (i=0; i<=1000; i++) 则程序运行结果不变。()

选择题

- 5) 第 12 行改为 for(i=1000;i>1;i--):第 13 行改为 for(j=a[i]; j>1; j--) 输入数据为 5 21 23 3 34 44，则运行结果()
 A. 不变 B. 则输出 21 23 3 34 44 C. 无输出 D. 则输出 44 34 3 32 21
- 6) (4 分) 第 10 行改为 ++a[t] 或 a[t++]，则输入 5 1 2 3 4 5，输出结果为()
 A. 1 2 3 4 5 或 5 4 3 2 1 B. 1 2 3 4 5 或无输出
 C. 5 4 3 2 1 或 5 4 3 2 1 D. 5 4 3 2 1 或无输出

3、

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  const int maxn=500000, INF=0x3f3f3f3f;
4  int L[maxn/2+2], R[maxn/2+2];
5  void unknown(int a[],int n,int left,int mid,int right) {
6      int n1=mid-left , n2=right-mid;
7      for(int i=0; i<n1; i++)
8          L[i]=a[left+1];
9      for(int i=0; i<n2; i++)
10         R[i]=a[mid+i];
11     L[n1]=R[n2]=INF;
12     int i=0,j=0;
13     for(int k=left; k<right; k++) {
14         if(L[i]<=R[j])
15             a[k]=L[i++];

```

```

16         else
17             a[k]=R[j++];
18     }
19 }
20 void unknownsort(int a[],int n, int left,int right) {
21     if(left+1<right) {
22         int mid=(left+right)/2;
23         unknownsort(a,n,left ,mid);
24         unknownsort(a,n, mid,right);
25         unknown(a,n,left , mid, right);
26     }
27 }
28 int main() {
29     int a[maxn],n;
30     cin>>n;
31     for(int i=0; i<n; i++)    cin>>a[i];
32     unknownsort(a,n,0,n);
33     for(int i=0; i<n; i++) {
34         if(i) cout<<" ";
35         cout<<a[i];
36     }
37     cout<<endl;
38     return 0;
39 }

```

判断题

- 1) 第 13 行的<改为<=将不会改变运行结果。()
- 2) 第 21 行的<改为<=将不会改变运行结果。()
- 3) 此类排序方法是高效的但是不稳定。()
- 4) 第 4 行的 2 个 “+2” 都去掉将不会改变运行结果。()

选择题

- 5) 此题是哪种排序 ()
A.选择排序 B.桶排序 C.归并排序 D.堆排序
- 6) 此题用到了 () 思想
A.动态规划 B.分治 C.冒泡 D.贪心

三、完善程序（单选题，每题 3 分，共计 30 分）

1、田忌赛马，田忌每赢一次齐王的马就得 200 金币，当然输了就扣 200 金币，平局则金币数不变。

```

#include<bits/stdc++.h>
using namespace std;
int main() {
    int n;
    while(cin>>n&&n!=0) {
        int tj[1001], king[1001], count=0;

```

```

int tj_min=0, tj_max=n-1;
int king_min=0,king_max=n-1;
for(int i=0; i<n; i++) cin>>tj[i];
for(int i=0; i<n; i++) cin>>king[i];
sort(tj,tj+n);
sort(king, king+n);
while(n--) {
    if(tj[__(1)__]>king[__(2)__]) {
        count++;
        tj_max--;
        king_max--;
    }
    else if(tj[__(3)__]<king[__(4)__]) {
        count--;
        tj_min++;
        king_max--;
    }
    else
    {
        if(tj[tj_min]>king[king_min]) {
            count++;
            __(5__);
            __(6__);
        }
        else{
            if(__(7__))
                count--;
            tj_min++;
            __(8__);
        }
    }
}
cout<<count*200<<endl;
}
return 0;
}

```

选择题

1) (1)和(2)处填 ()

A.tj_max 和 king_max

B.tj_min 和 king_max

C.tj_min 和 king_min

D.tj_max 和 king_min

2) (3)和(4)处填 ()

A.tj_min 和 king_max

B.tj_min 和 king_min

C.tj_max 和 king_max

D.tj_max 和 king_min

3) (5)和(6)处填 ()

- A.tj_min-和 king_min++ B.tj_max++和 king_min++
 C.tj_min++和 king_min++ D.tj_max++和 king_min--
 4) (7) 处填 ()
 A.tj[tj_min]<king[king_max] B.tj[tj_min]>king[king_max]
 C.tj[tj_max]<king[king_max] D.tj[tj_min]>king[king_min]
 5) (8) 处填 ()
 A.king_max-- B.king_max++ C.king_min-- D.king_min++

2、寻路问题: N*N 矩阵, 其中 0 是表示可以走的, 1 表示无法走, 矩阵有二维数组表示, 走左上角是入口, 右下角是出口, 只能横着走和竖着走, 要求找出最短路径

```
#include<bits/stdc++.h>
using namespace std;
int mymax=10000;
int f[4][2]= {{-1,0},{1,0},{0,-1},{0,1}};
int a[20][20],v[20][20],v1[20][20];
int l=1;

int n; //矩阵的规模

bool check(int x1,int y1) {
    if(x1<0||x1>=n ||__(1)__)return false;
    if(a[x1][y1]==1||__(2)__)return false;
    return true;
}

void dfs(int x,int y) {
    if(x==n-1&&y==n-1) {
        if(l<mymax) {
            mymax=l ;
            memcpy(v1,v,sizeof(v1));
        }
        return ;
    }
    for(int k=0; k<4; k++) {
        int x1,y1;
        x1=x+__(3)__;
        y1=y+__(4)__;
        if(check(x1,y1)) {
            __(5)__;
            __(6)__;
            dfs(x1,y1);
            __(7)__;
            v[x1][y1]=0;
        }
    }
}
```

```

int main() {
    cin>>n;
    for(int i=0; i<n; i++) {
        for(int j=0; j<n; j++)
            cin>>a[i][j];
    }
    dfs(0, 0);
    int d=v1[n-1][n-1];
    int x=n-1,y=n-1;
    int k;
    int qn[400][2];
    qn[0][0]=n-1;
    qn[0][1]=n-1;
    for(k=1;; k++) {
        x=x-f[d][0];
        y=y-f[d][1];
        qn[k][0]=x;
        qn[k][1]=y;
        d=v1[x][y];
        if(x==0&&y==0)break;
    }
    for(int i=k; i>=0; i--)
        cout<<"(8)"<<" "<<"(9)"<<endl;
    return 0;
}

```

选择题

1) (1)和(2)处填 ()

A. $y1 \leq 0 \parallel y1 > n$ 和 $v[x1][y1] > 0$ B. $y1 < 0 \parallel y1 \geq n$ 和 $v[x1][y1] > 0$

C. $y1 > 0 \&\&y1 <= n$ 和 $v[x1][y1] > 0$ D. $y1 > 0 \&\&y1 < n$ 和 $v[x1][y1] > 0$

2) (3)和(4)处填 ()

A. $f[k][0]$ 和 $f[k][1]$ B. $f[k][1]$ 和 $f[k][0]$ C. $f[0][k]$ 和 $f[1][k]$ D. $f[1][k]$ 和 $f[0][k]$

3) (5)处填 ()

A. $v[x1][y1]=k+1;$ B. $v[x1][y1]=k;$ C. $v[x][y]=k;$ D. $v[x][y]=k+1;$

4) (6)和(7)处填 ()

A. $1++$ 和 $1--$ B. $k++$ 和 $k--$ C. $x1++$ 和 $x1--$ D. $y1++$ 和 $y1--$

5) (8)和(9)处填 ()

A. $qn[i][1]$ 和 $qn[1][2]$ B. $qn[i][0]$ 和 $qn[i][1]$

C. $qn[1][i]$ 和 $qn[2][1]$ D. $qn[0][i]$ 和 $qn[1][i]$