

入门级 CSP-J 第 8 套初赛模拟试题

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)

- IPv4 中,以下 IP 地址不合法的是()。
A. 255. 255. 255. 255 B. 0. 1. 1. 1 C. 1. 1. 1. 0 D. 1. 0. 0. 0
- 已知 A, B, C 是 3 个二进制数,符号 \wedge 表示逻辑与运算,符号 \vee 表示逻辑或运算。
若 $A = 1100\ 1101\ 0011$
 $B = 1100\ 0111\ 0110$
 $C = 0011\ 0110\ 1010$
则表达式 $(A \vee B) \wedge (A \vee C)$ 的值为()。
A. 1100 1110 0001 B. 0011 0010 1111 C. 1100 1111 0011 D. 1100 0111 0001
- Linux 下可执行文件的默认扩展名为()。
A. exe B. chm C. dll D. 都不是
- 八进制数 7042 转化为十六进制数是()。
A. 3521 B. F22 C. E22 D. 111000100010
- 以下排序算法中,不需要进行关键字比较操作的算法是()。
A. 基数排序 B. 冒泡排序 C. 堆排序 D. 直接插入排序
- 一个袋子中有 3 个蓝球,2 个红球,2 个黄球,则从中抽出三个球颜色各不相同的概率是多少?()。
A. 10/21 B. 13/33 C. 12/35 D. 3/7
- 定义 L 数:素数或者是回文数满足两者中任意一个条件的数。大于等于 10 并且小于等于 120 的“L 数”共有多少个?(注:回文数指从左到右读与从右到左读是相同的,如“121”“1331”;两个条件都成立也是 L 数,如“131”)()。
A. 34 B. 35 C. 36 D. 37
- 定义一颗有根树的深度:根结点的深度为 0,其余结点的深度等于该结点的父亲结点的深度加 1。以下数字中哪一个可以作为一颗深度为 9 的完全二叉树的总节点数?()。
A. 511 B. 510 C. 1023 D. 1026
- 共 9 个互不相同的数,它们的最大公约数是 2021 的一个大于 1 的因子(6 有 2、3、6 这三个大于 1 的因子,因子可以包含自身),且这 9 个数的和小于等于 2021,则这 9 个数的和是多少?()。
A. 1849 B. 1935 C. 2021 D. 1927
- 以下哪位科学家被称为“博弈论之父”,“现代计算机之父”? ()。
A. 图灵 B. 冯诺依曼
C. 塔扬 D. 比尔盖茨
- 设栈 S 和队列 Q 初始状态为空,元素 a_1, a_2, \dots, a_6 依次通过栈 S, 一个元素出栈后就进入队列 Q, 若出队的顺序分别是 $a_2, a_4, a_3, a_6, a_5, a_1$, 则栈 S 的容量至少是()。
A. 2 B. 3 C. 4 D. 5
- 对有序数组 $\{5, 13, 19, 21, 37, 56, 64, 75, 88, 92, 100\}$ 进行二分查找,等概率的情况下查找成功的平均查找长度(平均比较次数)是()。
A. 35/11 B. 34/11 C. 3 D. 32/11

13. 一个 n 个顶点的强连通图最少有几条边? ()。
- A. n B. $n+1$ C. $n-1$ D. $n * (n-1)$
14. 在 1 和 2015 之间(包括 1 和 2015 在内)不能被 4、5、6 三个数任意一个数整除的数有几个? ()。
- A. 1035 B. 1105 C. 1075 D. 2000
15. 关于 Catalan 数 C_n , 下列说法中错误的是()。
- A. C_n 表示有 $n+1$ 个结点的不同形态的二叉树的个数
- B. C_n 表示含 n 对括号的合法括号序列的个数
- C. C_n 表示长度为 n 的入栈序列对应的合法出栈序列个数
- D. C_n 表示通过连接顶点而将 $n+2$ 边的凸多边形分成三角形的方法个数

二、阅读程序(程序输入不超过数组或字符串定义的范围:判断题正确填√,错误填×;除特殊说明外,判断题 1.5 分,选择题 3 分,共计 40 分)

1.

```

01 #include<bits/stdc++.h>
02 using namespace std;
03 int p;
04 void fun(int &x,int &y);
05 void func(int &x,int &y){
06     if(y>x) return;
07     x--;y/=2;
08     fun(x,y);
09 }
10 void fun(int &x,int &y){
11     if(x==1) return;
12     x/=2;y+=p;
13     func(x,y);
14 }
15 int main(){
16     int x,y;
17     cin>>x>>y>>p;
18     fun(x,y);
19     cout<<x<<' '<<y;
20     return 0;
21 }
    
```

●判断题

- (1) 将第 04 行的 & 去除后,程序仍能通过编译。()
- (2) 读入的 x, y, p 为 int 范围内任意值时程序均能完成运行。()
- (3) 若 $x=1$ 时,输出的 x, y 与输入的一致。()
- (4) 输出的 x 必然小于等于输入的 x 。()

●选择题

- (5) 输入为 7 33 2 时,输出为()。
- A. 4 31 B. 4 35 C. 3 31 D. 3 35

(6) 输入为 33 7 2 时, 输出为()。

A. 5 3

B. 3 5

C. 6 4

D. 4 6

2.

```
01 #include<iostream>
02 using namespace std;
03 const int maxn=105;
04 int n,a[maxn],b[maxn];
05 int main()
06 {
07     cin>>n;
08     int x;
09     for (int i=1;i<=n;i++) {
10         cin>>x;
11         a[i]=b[i]=x;
12     }
13
14     for (int i=1;i<=n;i++)
15         for (int j=i+1;j<=n;j++) {
16             if (a[i]>a[j]) swap(a[i],a[j]);
17             if (b[i]<b[j]) swap(b[i],b[j]);
18         }
19
20     for (int i=1;i<=n;i++) cout<<a[i]<<" ";
21     cout<<"\n";
22     for (int i=1;i<=n;i++) cout<<b[i]<<" ";
23     cout<<"\n";
24     return 0;
25 }
```

●判断题

- (1) 若输入的 $X[1], X[2], \dots, X[N]$ 中有相同的数, 程序会陷入死循环。()
- (2) 当且仅当输入的 $X[1], X[2], \dots, X[N]$ 全部相同时输出的两行结果相同。()
- (3) 该算法的原理是基数排序。()

●选择题

- (4) 若输入的 $X[1], X[2], \dots, X[N]$ 互不相同, 则下列说法正确的是()。

A. 输出的两行结果相同

B. 将输出的第一行结果整体翻转后, 将与第二行相同

C. 将输出的第一行结果的第一项与最后一项交换后, 将与第二行相同

D. 以上说法都不正确

- (5) 下列说法不正确的是()。

A. 输出的第一行即为将 $X[1], X[2], \dots, X[N]$ 从小到大排序后得到的结果

B. 输出的第二行即为将 $X[1], X[2], \dots, X[N]$ 从大到小排序后得到的结果

C. 若将“ $a[i] > a[j]$ ”改为“ $a[i] \geq a[j]$ ”,则程序输出无变化

D. 不存在时间复杂度更优的能与本程序达到相同目的的算法

(6) 该程序的时间复杂度为()。

A. $O(n)$

B. $O(n \log n)$

C. $O(n^2)$

D. $O(n \sqrt{n})$

3.

```
01 #include<bits/stdc++.h>
02 using namespace std;
03 int main() {
04     int num=0;
05     cin>>num;
06     //保证 num>=100,且在 int 范围内
07     int max_primedivisor=0;
08     int cnt=1;
09     for (int i=2;i*i<=num;i++){
10         if (num%i==0){
11             int tmp=1;
12             while (num%i==0) num/=i,tmp++;
13             max_primedivisor=max(max_primedivisor,i);
14             cnt*=tmp;
15         }
16     }
17     max_primedivisor=max(max_primedivisor,num);
18     if (num>1) cnt*=2;
19     cout<<max_primedivisor<<" "<<cnt<<"\n";
20     return 0;
21 }
```

●判断题

(1) 代码中 $\text{max_primedivisor} = \max(\text{max_primedivisor}, \text{num})$; 这句话去掉对答案没有影响。
()

(2) 当读入的 $\text{num} = p * q$, 其中 $p < q$, 且 p, q 为质数, 则 for 循环中 i 遍历到 q 时退出循环。
()

●选择题

(3) 该算法的最坏时间复杂度为()。

A. $O(\log \text{num})$

B. $O(\sqrt{\text{num}})$

C. $O(\text{num})$

D. $O(\text{num} \sqrt{\text{num}})$

(4) 当读入 2021 时输出为()。

A. 43 2

B. 43 4

C. 47 2

D. 47 4

(5) 当读入的数 $\text{num} = p * p * p * q * q * r * r * s * t$ 时, 其中 $p < q < r < s < t$, 且 p, q, r, s, t 均为质数, 则输出的第二个数()。

A. 不确定

B. 9

C. 12

D. 144

(6) 在最好的情况下, 时间复杂度为()。

A. $O(\log \text{num})$

B. $O(\sqrt{\text{num}})$

C. $O(\text{num})$

D. $O(\text{num} \sqrt{\text{num}})$

三、完善程序(单选题,每小题 3 分,共计 30 分)

1. (电鼠与方阵)有一个 $n * n$ ($2 \leq n \leq 5000$) 的方阵,其中每个方格有一个电力值。小 Y 可以在这个方阵中得到电力,方法就是在一些方格放上电鼠来吸收电力,这样就可以获得这些方格上的电力。不过放的电鼠须要遵循两个规则:1. 一个方格最多只能放一只电鼠;2. 所有 $2 * 2$ 的子矩阵(共有 $(n-1) * (n-1)$ 个)必须恰好包含两只电鼠。小 Y 用了一个程序求出了能获得的最大总电力值。试补全程序。

```

01 #include<bits/stdc++.h>
02 using namespace std;
03 const int N=5100;
04 int a ①;
05 int main() {
06     int n,ans1=0,ans2=0;
07     scanf("%d",&n);
08     for (int i=1;i<=n;i++)
09         for (int j=1;j<=n;j++)
10             scanf("%d",&a[i][j]);
11     for (int i=1;i<=n;i++) {
12         int odd=0,even=0;
13         for (int j=1;j<=n;j++) {
14             int x= ②;
15             if (j & 1) odd+=x;else even+=x;
16         }
17         ans1+=max(odd,even);
18     }
19     for (int i=1;i<=n;i++) {
20         int odd=0,even=0;
21         for (int j=1;j<=n;j++) {
22             int x= ③;
23             if ( ④ ) even+=x;else odd+=x;
24         }
25         ans2+=max(odd,even);
26     }
27     printf("%d\n", ⑤);
28     return 0;
29 }

```

(1)①处应该填()。

- A. [N][2] B. [2][N] C. [N][1100] D. [5100][5100]

(2)②处应该填()。

- A. a[j][i] B. a[i][j]

C. $a[i+j][(i+j)\&1]$

D. $a[(i+j)\&1][i+j]$

(3) ③处应该填()。

A. $a[j][i]$

B. $a[i][j]$

C. $a[i+j][(i+j)\&1]$

D. $a[(i+j)\&1][i+j]$

(4) ④处应该填()。

A. $j \& 1$

B. $j | 1$

C. $!(j \& 1)$

D. $!(j | 1)$

(5) ⑤处应该填()。

A. $\max(ans1, ans2)$

B. $\min(ans1, ans2)$

C. $ans1+ans2$

D. $\max(ans1, ans2) - \min(ans1, ans2)$

2. (排列) 给定一个 $1 \sim n$ 的排列 A, 你需要给出一个 $1 \sim n$ 的排列 B, 使得排列 B 的字典序的值最小。输出字典序最小的排列 B。

输入两行, 第一行一个正整数 n,

第二行 n 个整数表示排列 A。

提示: 将问题分为 n 是奇数和 n 是偶数考虑, 贪心处理。

试补全程序。

```
01 #include<bits/stdc++.h>
02 using namespace std;
03 int A[1000010]; int B[1000010]; int C[1000010];
04 int main() {
05     int n; scanf("%d", &n);
06     for(int i=1; i<=n; i++) scanf("%d", &A[i]);
07
08     if( ① ) {
09         int p1=0; int p2= ② ;
10         for(int i=1; i<=n; i++) {
11             if(A[i]>n/2) {
12                 B[i]=++p1;
13             } else {
14                 B[i]=++p2;
15             }
16         }
17     } else {
18         int p1=0; int p2=n/2;
19         for(int i=1; i<=n; i++) {
20             if(A[i]> ③ ) {
21                 B[i]=++p1;
22             } else {
23                 B[i]=++p2;
24             }
25         }
```

26

27 `p1=0;p2=n/2+1;`28 `for(int i=1;i<=n;i++){`29 `if(A[i]>= ④){`30 `C[i]=++p1;`31 `}else{`32 `C[i]=++p2;`33 `}`34 `}`

35

36 `int flag=0;`37 `for(int i=1;i<=n;i++){`38 `if(B[i]<C[i]){flag=1;break;}`39 `if(B[i]>C[i]){flag=2;break;}`40 `}`41 `if(flag!= ⑤) swap(B,C);`42 `}`43 `for(int i=1;i<n;i++)printf("%d ",B[i]);`44 `printf("%d\n",B[n]);`45 `return 0;`46 `}`

(1) ①处应填()。

A. $n\%2==0$ B. $n\%2==1$ C. $n==1$ D. $n==2$

(2) ②处应填()。

A. p1

B. $n/2-1$ C. $n/2$ D. $n/2+1$

(3) ③处应填()。

A. p1

B. $n/2-1$ C. $n/2$ D. $n/2+1$

(4) ④处应填()。

A. p1

B. $n/2-1$ C. $n/2$ D. $n/2+1$

(5) ⑤处应填()。

A. 0

B. 1

C. 2

D. 3