CSP-J 初赛模拟卷（第 09 套）

入门级 C++ 语言试题

一、单项选择题（共 **15** 题，每题 **2** 分，共计 **30** 分）

1．以下不属于面向对象程序设计语言的是（ ）。

A．C++　　B．Python　　C．Java　　D．C

2．在内存储器中每个存储单元都被赋予一个唯一的序号，称为（ ）

A．地址　　B．序号　　C．下标　　D．编号

3．广域网的英文缩写是（ ）。

A．LAN　　B．WAN　　C．MAN　　D．LNA

4．现有一张分辨率为 2048×1024 像素的 32 位真彩色图像。请问要存储这张图像，需要多大的存储空间？（ ）。

A．16MB　　B．4MB　　C．8MB　　D．2MB

5．对于入栈顺序为 a,b,c,d,e 的序列，下列（ ）不是合法的出栈序列。

A．a，b，c，d，e　　B．e，d，c，b，a

C．b，a，c，d，e　　D．c，d，a，e，b

6．设有100个已排好序的数据元素，采用折半查找时，最大比较次数为（ ）。

A．7　　B．10　　C．6　　D．8

7．有 10 个顶点的无向图至少应该有（ ）条边才有可能构成连通图。

A．9　　B．10　　C．11　　D．12

8．10 个三好学生名额分配到 7 个班级，每个班级至少有一个名额，一共有（ ）种不同的分配方案。

A．84　　B．72　　C．56　　D．504

9．独根树的高度为 1,具有 61 个结点的完全二叉树的高度为（ ）。

A．5　　B．6　　C．7　　D．8

10．把8个同样的球放在5个同样的袋子里，允许有的袋子空着不放，问共有多少种不同的分法？（　） 提示：如果8个球都放在一个袋子里，无论是哪个袋子，都只算同一种分法。

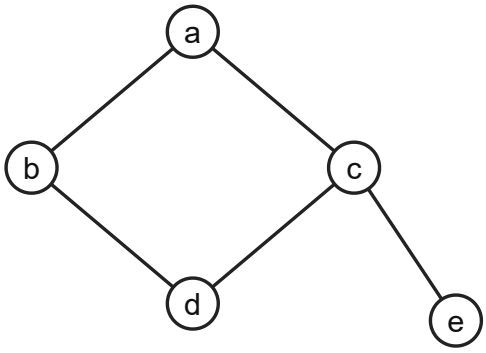
A．22　　B．24　　C．18　　D．20

11．在数据压缩编码中的哈夫曼编码方法，在本质是一种（　）的策略。

A．枚举　　B．贪心　　C．递归　　D．动态规划

12．考虑如下递归算法，则调用solve(7)得到的返回结果为（　）。

```
01 solve(n)
02 if n<=1 return 1
03 else if n>=5 return n*solve(n-2)
04 else return n*solve(n-1)
```



A．105　　B．840　　C．210　　D．420

13．以a为起点，对右边的无向图进行深度优先遍历，则b,c,d,e四个点中有可能作为最后一个遍历到的点个数为（　）。

A．1　　B．2　　C．3　　D．4

14．有四个人要从A点坐一条船过河到B点，船一开始在A点。该船一次最多可坐两个人。已知这四个人中每个人独自坐船的过河时间分别为1，2，4，8，且两个人坐船的过河时间为两人独自过河时间的较大者。则最短（　）时间可以让四个人都过河到B点（包括从B点把船开回A点时间）。

A．14　　B．15　　C．16　　D．17

15．319和377的最大公约数是（　）。

A．27　　B．33　　C．29　　D．31

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 **×**；共 **18** 小题，共计 **39.5** 分，各小题分值见题目）

(1)

```
01 #include <cstdio>
02 #include <cstring>
03 using namespace std;
04 char st[100001];
05 int main() {
06     scanf("%s", st);
07     int n = strlen(st);
08     for (int i = 1; i <= n; ++i) {
09         if (n % i == 0) {
10             char c = st[i - 1];
11             if (c >= 'a')
12                 st[i - 1] = c - 'a' + 'A';
13         }
14     }
15     printf("%s", st);
16     return 0;
17 }
```

判断题

16. (1.5分) 输入的字符串只能由小写字母或大写字母组成。（ ）

17. (1.5分) 若将第8行的“i=1”改为“i=0”，程序运行时会发生错误。（ ）

18. (1.5分) 若将第8行的“ $i <= n$ ”改为“ $i * i <= n$ ”，程序运行结果不会改变。（ ）

19. (1.5分) 若输入的字符串全部由大写字母组成，那么输出的字符串就跟输入的字符串一样。（ ）

单选题

20. (3分) 若输入的字符串长度为18,那么输入的字符串跟输出的字符串相比，至多有（ ）个字符不同。

A. 18　　B. 6　　C. 10　　D. 1

21. (3分) 若输入的字符串长度为（ ），那么输入的字符串跟输出的字符串相比，至多有36个字符不同。

A. 36　　B. 100000　　C. 1　　D. 128

(2)

```
01 #include <cstdio>
02 using namespace std;
03 int n;
04 int a[100];
05
06 int main() {
07     scanf("%d", &n);
08     for (int i = 1; i <= n; ++i)
09         scanf("%d", &a[i]);
10     int ans = 1;
11     for (int i = 1; i <= n; ++i) {
12         if (i > 1 && a[i] < a[i - 1])
13             ans = i;
14         while (ans < n && a[i] >= a[ans + 1])
15             ++ans;
16         printf("%d\n", ans);
17     }
18     return 0;
19 }
```

判断题

22. (1分) 第16行输出ans时，ans的值一定大于i。（ ）

23. (1分) 程序输出的ans小于等于n。（ ）

24. (1.5分) 若将第12行的 $<$ 改为 `!=` ，程序输出的结果不会改变。（ ）

25. (1.5分) 当程序执行到第16行时，若 $\mathrm{ans}-\mathrm{i}>2$，则 $\mathrm{a}[i+1]\le \mathrm{a}[i]$。（ ）

单选题

26. (3分) 若输入的a数组是一个严格单调递增的数列，此程序的时间复杂度 （ ）

A. O(log n)　　B. O(n^2)　　C. O(n log n)　　D. O(n)

27. (4分) 最坏情况下，此程序的时间复杂度是 （ ）

A. O(n^2)　　B. O(log n)　　C. O(n)　　D. O(n log n)

(3)

```
01 #include <stdio.h>
02
03 #define n 100000
04 #define N n+1
05
06 int m;
07 int a[N], b[N], c[N], d[N];
08 int f[N], g[N];
09
10 void init()
11 {
12     f[1] = g[1] = 1;
13     for (int i = 2; i <= n; i++) {
14         if (!a[i]) {
15             b[m++] = i;
16             c[i] = 1, f[i] = 2;
17             d[i] = 1, g[i] = i + 1;
18         }
19         for (int j = 0; j < m && b[j] * i <= n; j++) {
20             int k = b[j];
21             a[i * k] = 1;
22             if (i % k == 0) {
23                 c[i * k] = c[i] + 1;
24                 f[i * k] = f[i] / c[i * k] * (c[i * k] + 1);
25                 d[i * k] = d[i];
26                 g[i * k] = g[i] * k + d[i];
27                 break;
28             }
29             else {
30                 c[i * k] = 1;
31                 f[i * k] = 2 * f[i];
32                 d[i * k] = g[i];
33                 g[i * k] = g[i] * (k + 1);
34             }
35         }
36     }
37 }
38
39 int main()
40 {
41     init();
```

```
42
43      int x;
44      scanf("%d", &x);
45      printf("%d %d\n", f[x], g[x]);
46      return 0;
47  }
```

假设输入的x是不超过1000的自然数，完成下面的判断题和单选题：

判断题

28. (1.5分) 若输入不为“1”，把第12行删去不会影响输出的结果。（ ）

29. (2分) 第24行 $f[i*k]=f[i]/c[i*k]*(c[i*k]+1)$ 中的“$f[i]/c[i*k]$”可能存在无法整除而向下取整的情况。（ ）

30. (2分) 在执行完init()后，做组不是单调递增的，但g数组是单调递增的。（ ）

单选题

31. (3分) init函数的时间复杂度为（ ）。

A. O(n)　　B. O(nlogn)　　C. O(n√n)　　D. O(n²)

32. (3分) 在执行完init()后，f[1], f[2], f[3]…..f[100]中有（ ）个等于2。

A. 23　　B. 24　　C. 25　　D. 26

33. (4分) 当输入”1000”时，输出为（ ）。

A. “15 1340”　　B. “15 2340”　　C. “16 2340”　　D. “16 1340”

三、完善程序（程序输入不超过数组或字符串定义的范围；共 **10** 小题，共计 **29** 分，各小题分值见题目）

(1)

完善程序：

（快速幂）请完善下面的程序，该程序使用分治法求 $x^p \mod m$ 的值。

输入:三个不超过 10000 的正整数 x, p, m。

输出：$x^p \mod m$ 的值。

提示:若 p 为偶数，$x^p = (x^2)^{p/2}$；若 p 为奇数，$x^p = x * (x^2)^{(p-1)/2}$。

```
01 #include<iostream>
02 using namespace std;
03 int x, p, m, i, result;
04 int main(){
05     cin >> x >> p >> m;
06     result = __(1)__;
07     while (__(2)__){
08         if (p % 2 == 1)
09             result = __(3)__;
10         p /= 2;
11         x = __(4)__;
12     }
13     cout << __(5)__ << endl;
14     return 0;
15 }
```

34. (2分) (1)处应填：_____

35. (3分) (2)处应填：_____

36. (3分) (3)处应填：_____

37. (3分) (4)处应填：_____

38. (3分) (5)处应填：_____

(2)

（Josephus 问题）有 n个人围成一个圈，依次标号 0 至 n-1。从 0 号开始，依次 0, 1, 0, 1, … 交替报数，报到 1 的人会离开，直至圈中只剩下一个人。求最后剩下人的编号。 试补全模拟程序。

```
01 #include <stdio.h>
02
03 const int MAXN = 1000000;
04 int F[MAXN];
05
06 int main(){
07     int n;
08     scanf("%d", &n);
09     int i = 0, p = 0, c = 0;
10     while( __(1)__){
```

```
11          if (F[i] == 0){
12              if ( __(2)__ ){
13                  F[i] = 1;
14                  __(3)__;
15              }
16              __(4)__;
17          }
18          __(5)__;
19      }
20      int ans = -1;
21      for (int i = 0; i < n; i++)
22          if (F[i] == 0)
23              ans = i;
24      printf("%d\n", ans);
25      return 0;
26  }
```

39. (3分) (1)处应填

A. i < n　　B. c < n　　C. i < n - 1　　D. c < n - 1

40. (3分) (2)处应填

A. i % 2 == 0　　B. i % 2 == 1　　C. p　　D. !p

41. (3分) (3)处应填

A. i++　B. i = (i+1) % n　C. c++　D. p ^= 1

42. (3分) (4)处应填

A. i++　B. i = (i+1) % n　C. c++　D. p ^= 1

43. (3分) (5)处应填

A. i++　B. i = (i+1) % n　C. c++　D. p ^= 1