CSP-J 初赛模拟卷（第 18 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．已知十进制中的18在x的进制中的表示为24，即(18)_10=(24)_x，则这个x进制为（ ）进制。

A．9　　B．8　　C．7　　D．6

2．根据网址的域名 http://www.jiangsu.gov.cn/，可以判断该网站是（ ）类型的网站。

A．商业机构　　B．军事机构　　C．政府机构　　D．组织机构

3．与计算机硬件关系最密切的软件是（ ）。

A．编译程序　B．操作系统　C．游戏程序　D．数据库管理程序

4．下列程序段执行后s的值为（ ）。

```
int i = 1, s = 0;
while (i++)
    if (!(i % 3)) break;
    else s += i;
```

A．6　　B．3　　C．2　　D．以上均不是

5．将19分解成3个不重复数字(1~9)之和(不计顺序)的方法有（ ）种。

A．6　　B．5　　C．4　　D．3

6．甲、乙、丙三位同学选修课程，在四门课程中，甲选修两门，乙、丙各选修三门，则不同选修方案共有（ ）种。

A．36　　B．48　　C．64　　D．96

7．已知某二叉树的先序遍历序列是 ABDCE，中序遍历序列是 BDAEC，则该二叉树的后序遍历为（ ）。

A．BDECA　　B．DBECA　　C．DBCEA　　D．BDCEA

8．计算机启动时，可以通过存储在（ ）中的引导程序引导操作系统。

A．ROM　　B．RAM　　C．Cache　　D．CPU

9．表达式a+b*c-(d+e)的前缀形式是（　）

A．abc*+de+-　　B．-+*abc+de　　C．-+a*bc+de　　D．abcde*++-

10．小军在家玩关灯游戏，小军家的灯有三种颜色分别是白、黄、红。按1下白灯亮，按2下灯灭，按3下黄灯亮，按4下灯灭，按5下红灯亮，按6下灯灭，再按又是白灯亮，以此循环。当按到49次和100次时灯的状态是（　）。

A．灯灭，灯灭　B．红灯亮，灯灭　C．白灯亮，红灯亮　D．白灯亮，灯灭

11．704与2048的最小公倍数是（　）。

A．45 056　　B．22 528　　C．180 224　　D．90 112

12．在（　）的情况下，逻辑 $A \vee B$ 运算的结果是逻辑“0”。

A．A和B全部是1　　B．A和B任一是1　　C．A和B全部是0　　D．A和B任一是0

13．小明夫妇请了小刚夫妇和小伟夫妇来他们家玩扑克。这种扑克游戏有一种规则：夫妇两人不能一组。小明和小红一组，小刚的队友是小伟的妻子，琳达的丈夫和小丽一组。那么这三对夫妇分别为（　）

A．小明-小丽，小刚-琳达，小伟一小红　B．小明-琳达，小刚-小红，小伟-小丽
C．小明-小丽，小刚-小红，小伟一琳达　D．小明-小红，小刚-小丽，小伟-琳达

14．4人过桥，每人单独过桥分别需要用时1分、2分、5分、10分，过桥需要灯(只有一盏)，一次只能2人一起过(意味着需要有人送灯回来)，过桥时间以用时多的人为准，则4人全部过桥时间最少需要（　）分。

A．15　　B．16　　C．17　　D．18

15．2000 年,华人学者姚期智因在计算理论(包括伪随机数生成、密码学与通信复杂度)方面的突出成就而荣获（　）

A．图灵奖　　B．奥斯卡奖　　C．诺贝尔奖　　D．普利策奖

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 **37.5** 分，各小题分值见题目）

(1)

```
01 #include<iostream>
02 using namespace std;
03 int main()
04 {
05     const int SIZE=10;
06     int height[SIZE],num[SIZE],n,ans;
07     cin>>n;
08     for(int i=0;i<n;i++){
09         cin>>height[i];
10         num[i]=1;
11         for(int j=0;j<i;j++){
12             if((height[j]<height[i])&&(num[j]>=num[i]))
13                 num[i]=num[j]+1;
14         }
15     }
16     ans=0;
17     for(int i=0;i<n;i++){
18         if(num[i]>ans) ans=num[i];
19     }
20     cout<<ans<<endl;
21     return 0;
22 }
```

判断题

16. (1.5分) 如果height数组中的输入有负数，则程序会出错。（ ）

17. (1.5分) 程序输出的 ans 小于或等于n。 （ ）

18. (1.5分) 将12行num[j]>=num[i]”改为“num[j]>num[i]”，程序的输出结果不会改变。（ ）

19. (1.5分) 将18行“num[i]>ans”改为“num[i]>=ans”，程序的输出结果不会改变。（ ）

单选题

20. (3分) 若输入的数据为 10 1 1 1 1 1 1 1 1 1 则程序的输出结果是（ ）。

A. 4　　B. 3　　C. 2　　D. 1

21. (3分) 若输人的数据为 10 3 2 5 11 12 7 4 10 15 6 因程序的输出结果是（ ）。

A. 4　　B. 5　　C. 6　　D. 7

(2)

```
#include<iostream>
using namespace std;
int n,m,i,j,p,k;
int a[100],b[100];
int main()
{
    cin>>n>>m;
    a[0]=n;i=0;p=0;k=1;
    do{
        for(j=0;j<i;j++)
            if(a[i]==a[j])
            {
                p=1;k=j;break;
            }
        if(p) break;
        b[i]=a[i]/m;
        a[i+1]=a[i]%m*10;
        i++;
    }while(a[i]!=0);
    cout<<b[0]<<".";
    for(j=1;j<k;j++) cout<<b[j];
    if(p) cout<<"(";
    for(j=k;j<i;j++) cout<<b[j];
    if(p) cout<<") ";
    cout<<endl;
    return 0;
}
```

判断题

22. (1.5分) 程序输入的n和m不能相等。（ ）

23. (1.5分) 程序输人的m不能等于0。（ ）

24. (1.5分) 第9~19行的do…while循环一共有2个出口。（ ）

25. (1.5分) 数组a和b中的数值都小于或等于n。（ ）

单选题

26. (3分) 若输入数据为“11 8”,则输出结果为（ ）

A. 0.375　　B. 0.(375)　　C. 1.375　　D. 1.(375)

27. (3分) 若输人数据为“5 13”，则输出结果为（ ）。

A. 0.386514　　B. 0.(386514)　　C. 0.384615　　D. 0.(384615)

(3)

```
01 #include <iostream>
02 using namespace std;
03 const int V = 101;
04 int n, m, ans, e[V][V];
05 bool visited[V];
06 void dfs(int x, int len)
07 {
08     int i;
09     visited[x] = true;
10     if (len > ans) ans = len;
11     for (i = 1; i <= n; i++)
12         if ((!visited[i]) && (e[x][i] != -1))
13             dfs(i, len + e[x][i]);
14     visited[x] = false;
15 }
16 int main()
17 {
18     int i, j, a, b, c;
19     cin >> n >> m;
20     for (i = 1; i <= n; i++)
21         for (j = 1; j <= n; j++)
22             e[i][j] = -1;
23     for (i = 1; i <= m; i++) {
24         cin >> a >> b >> c;
25         e[a][b] = c;
26         e[b][a] = c;
27     }
```

```
28      for (i = 1; i <= n; i++)
29          visited[i] = false;
30      ans = 0;
31      for (i = 1; i <= n; i++)
32          dfs(i, 0);
33      cout << ans << endl;
34      return 0;
35  }
```

输入表示一个没有重边、自环的无向图，顶点编号为 1 到 n，且 $1 \le n \le 100$、边权为正整数。

判断题

28. (1.5分) 第 19 行的输入中，如果满足 $m = n(n-1)/2$，则第 20～22 行的初始化可以省略。（ ）

29. (1.5分) 将第 31 行改为 `for (i = n; i >= 1; i--)`，程序结果不受影响。（ ）

单选题

30. (1.5分) 若输入如下，则程序输出为（ ）。

```
4 2
1 2 1
3 4 1
```

A. 8　　B. 4　　C. 2　　D. 1

31. (3分) 若输入如下，则程序输出为（ ）。

```
4 6
1 2 1
2 3 1
3 4 1
4 1 1
1 3 1
2 4 1
```

A. 1　　B. 3　　C. 5　　D. 7

32. (3分) 若输入如下，则程序输出为（　）。

```
4 3
1 2 10
2 3 20
3 1 30
```

A. 30　　B. 50　　C. 90　　D. 120

33. (3分) 若输入如下，则程序输出为（　）。

```
4 6
1 2 10
2 3 20
3 4 30
4 1 40
1 3 50
2 4 60
```

A. 60　　B. 100　　C. 150　　D. 200

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 **30** 分，各小题分值见题目）

(1)

(高精度计算)由于计算机运算的数据范围表示有一定限制，如整型int表达范围是(-2^{31}~$2^{}$31-1)，unsigned long(无符号整数)是(0~2^32-1)，都约为几十亿，因此在计算位数超过十几位的数时，不能采用现有类型，只能自己编程计算。 高精度计算通用方法：高精度计算时一般用一个数组存储一个数，数组的一个元素对应于数的一位，将数由低位到高位依次存储在数组下标对应的由低到高的位置上。另外，申请数组大小时，一般考虑了最大的情况，在很多情况下表示有富裕，即高位有很多0，可能造成无效的运算和判断，因此一般利用一个整型数据存储该数的位数。下面的程序是一个高精度整数的加法运算，请补充完整程序。

```
01 #include <iostream>
02 #include <cstring>
03 using namespace std;
04
05 struct HugeInt{
06     int len;
07     int num[100002];
```

```
};
HugeInt a, b, w;
char c[100001],d[100001];
void Scan_HugeInt(){
    cin>>c;
    cin>>d;
    a.len=strlen(c);
    b.len=strlen(d);
    for(int i=0; i<a.len; i++)
        __(1)__;
    for(int i=0; i<b.len; i++)
        __(2)__;
}
void Plus(){
    w.len=max(a.len,b.len);
    for(int i=1; i<=w.len; i++) {
w.num[i] +=__(3)__;
w.num[i+1] +=__(4)__;
w.num[i]%=10;
}
if(__(5)__)
    w.len ++;
}

int main() {
    Scan_HugeInt();
    Plus();
    for(int i=w.len; i>=1; i--)
        cout <<w.num[i];
    cout <<endl;
    return 0;
}
```

34. (3分) (1)处应填

A. a.num[i]=c[i]-'0'　　B. a.num[a.len-i]=c[i]-'0'

C. a.num[a.len-i]=c[i]　　D. a.num[i]=c[i]

35. (3分) (2)处应填

A. b.num[i]=d[i]　　B. b.num[b.len-i]=d[i]

C. b.num[b.len-i]=d[i]-'0'　　D. b.num[i]=d[i]-'0'

36. (3分) (3)处应填

A. (a.num[i]+b.num[i])　　B. (a.num[i]+b.num[i])%10
C. (a.num[i]%10+b.num[i]%10)　　D. (a.num[i]+b.num[i]-10)

37. (3分) (4)处应填

A. w.num[i]/10　　B. w.num[i]　　C. w.num[i]%10　　D. w.num[i]-10

38. (3分) (5)处应填

A. w.num[w.len+1]>=0　　B. w.num[w.len+1]==0
C. w.num[w.len+1]!=0　　D. w.num[w.len +1]>1

(2)

(马走日)回溯算法实际上一个类似枚举的搜索尝试过程，主要是在搜索尝试过程中寻找问题的解，当发现已不满足求解条件时，就“回溯”返回，尝试其他路径。回溯法是一种选优搜索法按选向前索以达到目标。但当搜索到某一步时，若发现原先选择并优或达不到目标,则退回一步重新选择，这种走不通就退回去再走的技术称为回溯法，而满足回溯条件的某个状态的点称为回溯点。 马在中国象棋中以日字形规则移动。请编写一段程序，给定rxc大小的棋盘以及马的初始位置(m,n)，要求不能重复盘上的同一个点,计算马有多少途径可以遍历盘上的所有点。

```
01 #include<iostream>
02 using namespace std;
03 int r,c;
04 int cnt,tot;
05 int wayr[8]={2,2,1,-1,-2,-2,1,-1};
06 int wayc[8]={1,-1,2,2,1,-1,-2,-2};
07 bool mark[1001][1001];
08 bool check(int x, int y)
09 {
10     if(__(1)__) return true;
11     return false;
12 }
13 void search(int x,int y)
14 {
15     for(int i=0;i<8;i++)
16         if(__(2)__ && __(3)__)
17         {
18             mark[x+wayr[i]][y+wayc[i]]=true;
19             tot++;
```

```
            if(__(4)__) cnt++;
            search(__(5)__);
            tot--;
            mark[x+wayr[i]][y+wayc[i]]=false;
        }
}
int main()
{
    cnt=0;
    int m,n;
    cin>>r>>c>>m>>n;
    if(!check(m,n)) cout<<0<<endl;
    else if(r==1&&c==1) cout<<1<<endl;
    else
    {
        mark[m][n]=true;
        search(m,n);
        cout<<cnt<<endl;
    }
}
```

39. (3分) (1)处应填

A. x>=0 || y>=0 || x<r || y<c . B. x>=0 || y>=0 && x<r || y<e.
C. x>=0 && y>=0 && x<r && y<c D. x>=0&&y>=0|| x<r &&y<c

40. (3分) (2)处应填

A. check(wayr[i]，wayc[i]) B. check(x+wayr[i], y+wayc[i])
C. !check(x+wayr[i],y+wayc[i]) D. !check(wayr[I],wayc[i])

41. (3分) (3)处应填

A. mark[wayr[i]][wayc[i]] B. mark[x+wayr[i]][y+wayc[i]]
C. !mark[wayr[i]][wayc[i]] D. !mark[x+wayr[i]][y+wayc[i]]

42. (3分) (4)处应填

A. tot==r * c-1 B. tot-=r * c C. cnt==r * c-1 D. cnt==r * c

43. (3分) (5)处应填

A. wayr[i], wayc[i] B. x-wayr[i], y-wayc[i]
C. x-wayr[i], y+wayc[i] D. x+wayr[i], y+wayc[i]