



## C++ 七级 (样题)

### 1 单选题 (每题 2 分, 共 30 分)

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	A	B	C	D	B	D	B	A	C	C	D	C	D	B	B

第 1 题 如果下面代码输入整数为 10, 输出是 1, 则横线处填写? ( )

```
1 #include <iostream>
2 #include <string>
3 #include <cmath>
4 using namespace std;
5
6 int main()
7 {
8     int x;
9
10    cin >> x;
11
12    cout << _____ << endl;
13
14    return 0;
15 }
```

- A.  $\log_{10}(x)$
- B.  $\sin(x)$
- C.  $\exp(x)$
- D.  $\text{pow}(x,10)$

第 2 题 下面定义的函数用来求斐波那契数列的  $F(n)$ , 其中  $n \leq 10$ , 描述正确的是 ( )。

```
1 int fab(int n)
2 {
3     int f[n+1];
4
5     f[0]=0, f[1]=1;
6
7     for (int i=2; i<=n; i++)
8         f[i] = f[i-1] + f[i-2];
9
10    return f[n];
11 }
```

- A.  $f[0]$ 和 $f[1]$ 是递归终止条件
- B. 数组 $f$ 保存算法执行过程的状态
- C. 使用倍增法来求解
- D. 算法不能正常结束

第3题 下列关于C++语言中函数的叙述，正确的是（）。

- A. 函数定义前必须声明
- B. 函数调用前必须定义
- C. 函数调用时必须提供足够的实际参数
- D. 函数声明只能写在函数调用前

第4题 4个结点的简单有向图，最多可以有多少条边（）。

- A. 4
- B. 6
- C. 8
- D. 12

第5题 哈希表上可以执行的操作不包括（）

- A. 插入
- B. 排序
- C. 查找
- D. 删除

第6题 将关键码集合{100, 300, 500, 700, 800, 900}逐一保存在一个长度为100的哈希表中，选取哈希函数为 $\text{Hash}(\text{key})=\text{key}/100$ ，则800保存在表中的位置应该是（）。

- A. 5
- B. 6
- C. 7
- D. 8

第7题 定义double型常量 $\pi=3.14$ 和变量 $x$ ， $x$ 代表等边三角形边长，则该三角形的面积是（）。

- A.  $x*x*\sin(\pi/3)$
- B.  $x*x*\sin(\pi/3)/2$
- C.  $x*x*\cos(\pi/3)$
- D.  $x*x*\cos(\pi/3)/2$

第8题 动态规划将一个问题分解为一系列子问题后来求解。下面关于子问题的描述正确的是（）。

- A. 具有重叠子问题的性质
- B. 和分治法的子问题类似
- C. 不具有最优子结构的性质
- D. 问题的最优解可以由部分子问题的非最优解推导出来

第9题 阅读以下代码，visited起到的作用是（）。

```

1 #include <iostream>
2 #include <string>
3 #include <cmath>
4 using namespace std;
5
6 int visited[100], k=0;
7
8 void dfs(int graph[][100], int start, int vexnum)
9 {
10     visited[start]=++k;
11     cout << start << " ";
12     for (int i=0; i<vexnum; i++) {
13         if ((start!=i) && !visited[i]) {
14             dfs(graph, i, vexnum);
15         }
16     }
17 }

```

- A. 实现遍历过程
- B. 以广度优先的方式记录图中的顶点
- C. 存储深搜时节点的访问顺序
- D. 能够记录最短路径

第10题 下面函数尝试使用动态规划方法求出如下递推公式的函数，则横线处填写下列哪段代码可以完成预期功能？  
( )

$$C(n, m) = \begin{cases} 1 & \text{当 } n = 0 \text{ 或 } m = 0 \\ C(n-1, m-1) + C(n-1, m) & \text{当 } n > 0 \text{ 且 } m > 0 \end{cases}$$

```

1  int rec_C[MAXN][MAXM];
2  int C(int n, int m) {
3      for (int i = 0; i <= n; i++)
4          rec_C[i][0] = 1;
5      for (int j = 0; j <= m; j++)
6          rec_C[0][j] = 1;
7      _____ // 在此处填入代码
8          rec_C[i][j] = rec_C[i - 1][j - 1] + rec_C[i - 1][j];
9      return rec_C[n][m];
10 }

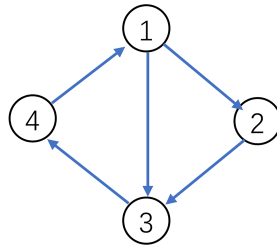
```

- A. for (int i = n; i >= 1; i--) for (int j = 1; j <= m; j++)
- B. for (int i = 1; i <= n; i++) for (int j = m; j >= 1; j--)
- C. for (int j = 1; j <= m; j++) for (int i = 1; i <= n; i++)
- D. for (int j = 1; j <= m; j++) for (int i = n; i >= 1; i--)

第 11 题 深度为4的完全二叉树，结点总数最少有多少个？（ ）

- A. 5
- B. 6
- C. 7
- D. 8

第 12 题 下面有向图中的数字表示顶点序号，则从1号顶点出发的BFS遍历的输出顶点序列可能是（ ）。

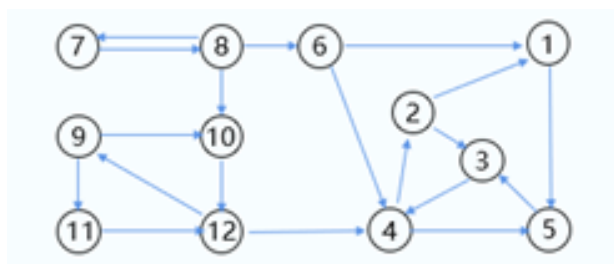


- A. 1 4 3 2
- B. 1 4 2 3
- C. 1 3 2 4
- D. 1 2 4 3

第 13 题 一个简单有向图有20个结点，假设图中已经存在300条边，请问增加多少条边可以成为完全图。（ ）

- A. 77
- B. 78
- C. 79
- D. 80

第 14 题 在下面的有向图中，强连通分量有多少个？（ ）



- A. 3
- B. 4

C. 5

D. 6

第 15 题 下面有关格雷码的说法，错误的是（ ）。

A. 在格雷码中，任意两个相邻的代码只有一位二进制数不同

B. 格雷码是一种唯一性编码

C. 在格雷码中，最大数和最小数只有一位二进制数不同

D. 格雷码是一种可靠性编码

## 2 判断题（每题 2 分，共 20 分）

题号	1	2	3	4	5	6	7	8	9	10
答案	×	×	√	√	×	√	×	√	×	×

第 1 题 定义变量double x=exp(-1)，则x<0为真。（ ）

第 2 题 假设x和y都是double型正数，如果说x比y大一个数量级，log(x/y)等于10。（ ）

第 3 题 如果double型变量x代表锐角对应的弧度角，则可以编程来确定sin(x)>cos(x)的近似区间。（ ）

第 4 题 pow(1,2)返回的结果是浮点数。（ ）

第 5 题 如果哈希表足够大，哈希函数确定后，不会产生冲突。（ ）

第 6 题 动态规划最终要推导出状态转移方程才能求解。（ ）

第 7 题 简单有向图的深搜结果和广搜结果一样。（ ）

第 8 题 判断图是否连通可以用深搜实现。（ ）

第 9 题 在C++中，可以使用二分法查找链表中的元素。（ ）

第 10 题 有些算法或数据结构在C/C++语言中使用指针实现，一个典型的例子就是链表。因此，链表这一数据结构在C/C++语言中只能使用指针来实现。（ ）

## 3 编程题（每题 25 分，共 50 分）

### 3.1 编程题 1

- 试题名称：迷宫统计
- 时间限制：1.0 s
- 内存限制：128.0 MB

#### 3.1.1 问题描述

在神秘的幻想大陆中，存在着  $n$  个古老而神奇的迷宫，迷宫编号从 1 到  $n$ 。有的迷宫之间可以直接往返，有的可以走到别的迷宫，但是不能走回来。玩家小杨想挑战一下不同的迷宫，他决定从  $m$  号迷宫出发。现在，他需要你帮助他统计：有多少迷宫可以直接到达  $m$  号迷宫， $m$  号迷宫可以直接到达其他的迷宫有多少，并求出他们的和。

需要注意的是，对于  $i(1 \leq i \leq n)$  号迷宫，它总可以直接到达自身。

### 3.1.2 输入描述

第一行两个整数  $n$  和  $m$ ，分别表示结点迷宫总数  $n$ ，指定出发迷宫的编号  $m$ 。

下面  $n$  行，每行  $n$  个整数，表示迷宫之间的关系。对于第  $i$  行第  $j$  列的整数，1 表示能从  $i$  号迷宫直接到达  $j$  号迷宫，0 表示不能直接到达。

### 3.1.3 输出描述

一行输出空格分隔的三个整数，分别表示迷宫  $m$  可以直接到达其他的迷宫有多少个，有多少迷宫可以直接到达  $m$  号迷宫，这些迷宫的总和。

### 3.1.4 样例输入1

```
1 | 6 4
2 | 1 1 0 1 0 0
3 | 0 1 1 0 0 0
4 | 1 0 1 0 0 1
5 | 0 0 1 1 0 1
6 | 0 0 0 1 1 0
7 | 1 0 0 0 1 1
```

### 3.1.5 样例输出1

```
1 | 3 3 6
```

### 3.1.6 样例解释

4号迷宫能直接到达的迷宫有3,4,6号迷宫，共3个。

能直接到达4号迷宫的迷宫有1,4,5号迷宫，共3个。

总和为6。

### 3.1.7 子任务

子任务编号	分值	$n$
1	30	$\leq 10$
2	30	$\leq 100$
3	40	$\leq 1000$

对于全部数据，保证有  $4 \leq n \leq 1000$ ， $1 \leq m \leq n$ 。

### 3.1.8 参考程序

```
1 #include<stdio>
2 #define M 101
3
4 int g[M][M];
5 int main(){
6     int n,m;
7     scanf("%d %d",&n,&m);
8     for(int i=1;i<=n;i++){
9         for(int j=1;j<=n;j++){
10             scanf("%d",&g[i][j]);
11         }
12     }
```

```

13 | int ans1 = 0,ans2 = 0;
14 | for(int i=1;i<=n;i++){
15 |     ans1+=g[i][m];
16 |     ans2+=g[m][i];
17 | }
18 | printf("%d %d %d",ans2,ans1,ans1+ans2);
19 | return 0;
20 | }

```

## 3.2 编程题 2

- 试题名称: 最长不下降子序列
- 时间限制: 1.0 s
- 内存限制: 128.0 MB

### 3.2.1 问题描述

小杨有一个包含  $n$  个节点  $m$  条边的有向无环图，其中节点的编号为 1 到  $n$ 。

对于编号为  $i$  的节点，其权值为  $A_i$ 。对于图中的一条路径，根据路径上的经过节点的先后顺序可以得到一个节点权值的序列，小杨想知道图中所有可能序列中最长不下降子序列的最大长度。

注：给定一个序列  $S$ ，其最长不下降子序列  $S'$  是原序列中的如下子序列：整个子序列  $S'$  单调不降，并且是序列中最长的单调不降子序列。例如，给定序列  $S = [11, 12, 13, 9, 8, 17, 19]$ ，其最长不下降子序列为  $S' = [11, 12, 13, 17, 19]$ ，长度为 5。

### 3.2.2 输入描述

第一行包含两个正整数  $n, m$ ，表示节点数和边数。第二行包含  $n$  个正整数  $A_1, A_2, \dots, A_n$ ，表示节点 1 到  $n$  的点权。之后  $m$  行每行包含两个正整数  $u_i, v_i$ ，表示第  $i$  条边连接节点  $u_i$  和  $v_i$ ，方向为从  $u_i$  到  $v_i$ 。

### 3.2.3 输出描述

输出一个正整数，表示该图中所有可能序列中最长不下降子序列的最大长度。

### 3.2.4 样例输入1

```

1 | 5 4
2 | 2 10 6 3 1
3 | 5 2
4 | 2 3
5 | 3 1
6 | 1 4

```

### 3.2.5 样例输出1

```

1 | 3

```

### 3.2.6 样例输入2

```
1 6 11
2 1 1 2 1 1 2
3 3 2
4 3 1
5 5 3
6 4 2
7 2 6
8 3 6
9 1 6
10 4 6
11 1 2
12 5 1
13 5 4
```

### 3.2.7 样例输出2

```
1 | 4
```

### 3.2.8 样例输入3

```
1 6 11
2 5 9 10 5 1 6
3 5 4
4 5 2
5 4 2
6 3 1
7 5 3
8 6 1
9 4 1
10 4 3
11 5 1
12 2 3
13 2 1
```

### 3.2.9 样例输出3

```
1 | 4
```

### 3.2.10 子任务

子任务编号	分值	n	$\max A_i$
1	30	$\leq 10^3$ 有向无环图为一链	$\leq 10$
2	30	$\leq 10^5$	$\leq 2$
3	40	$\leq 10^5$	$\leq 10$

对于全部数据，保证有  $1 \leq n \leq 10^5$ ,  $1 \leq m \leq 10^5$ ,  $1 \leq A_i \leq 10$ 。



### 3.2.11 参考程序

```
1 #include "bits/stdc++.h"
2 using namespace std;
3 const int N = 100010;
4 vector<int> g[N];
5 int n,m;
6 int a[N];
7 int d[N],dp[N][15];
8
9 int main(){
10     cin>>n>>m;
11     for(int i=1;i<=n;i++){
12         cin>>a[i];
13     }
14     for(int i=1;i<=m;i++){
15         int u,v;
16         cin>>u>>v;
17         g[u].push_back(v);
18         d[v]++;
19     }
20     queue<int> q;
21     for(int i=1;i<=n;i++){
22         if(!d[i]){
23             q.push(i);
24             dp[i][a[i]]=1;
25         }
26     }
27     while(!q.empty()){
28         int now = q.front();
29         q.pop();
30         for(auto i:g[now]){
31             for(int j=1;j<=10;j++){
32                 dp[i][j]=max(dp[i][j],dp[now][j]);
33             }
34             d[i]--;
35             if(!d[i]){
36                 for(int k=a[i];k>=1;k--){
37                     dp[i][a[i]]=max(dp[i][a[i]],dp[i][k]+1);
38                 }
39                 q.push(i);
40             }
41         }
42     }
43     int mx=1;
44     for(int i=1;i<=n;i++){
45         for(int j=1;j<=10;j++){
46             mx=max(mx,dp[i][j]);
47         }
48     }
49     cout<<mx<<"\n";
50 }
```