

# 第二届全国青少年信息学奥林匹克联赛初赛

## 普及组 C++语言试题

竞赛时间：2014 年 10 月 12 日 14:30~16:30

选手注意：

- 试题纸共有 8 页，答题纸共有 2 页，满分 100 分。请在答题纸上作答，写在试题纸上的  
一律无效。
- 不得使用任何电子设备（如计算器、手机、电子词典等）或查阅任何书籍资料。

一、单项选择题（共 20 题，每题 1.5 分，共计 30 分；每题有且仅有一个正确选项）

1. 以下哪个是面向对象的高级语言（ ）。  
A. 汇编语言      B. C++      C. Fortran      D. Basic
2. 1TB 代表的字节数量是（ ）。  
A. 2 的 10 次方      B. 2 的 20 次方      C. 2 的 30 次方      D. 2 的 40 次方
3. 二进制数 00100100 和 00010101 的和是（ ）。  
A. 00101000      B. 001010100      C. 01000101      D. 00111001
4. 以下哪一种设备属于输出设备（ ）。  
A. 扫描仪      B. 键盘      C. 鼠标      D. 打印机
5. 下列对操作系统功能的描述最为完整的是（ ）。  
A. 负责外设与主机之间的信息交换  
B. 负责诊断机器的故障  
C. 控制和管理计算机系统的各种硬件和软件资源的使用  
D. 将源程序编译成目标程序
6. CPU、存储器、I/O 设备是通过（ ）连接起来的。  
A. 接口      B. 总线      C. 控制线      D. 系统文件
7. 断电后会丢失数据的存储器是（ ）。  
A. RAM      B. ROM      C. 硬盘      D. 光盘

8. 以下哪一种是属于电子邮件收发的协议 ( )。
- A. SMTP                      B. UDP                      C. P2P                      D. FTP
9. 下列选项中不属于图像格式的是 ( )。
- A. JPEG 格式              B. TXT 格式              C. GIF 格式              D. PNG 格式
10. 链表不具有的特点是 ( )。
- A. 不必事先估计存储空间                      B. 可随机访问任一元素  
C. 插入删除不需要移动元素                      D. 所需空间与线性表长度成正比
11. 下列各无符号十进制整数中，能用八位二进制表示的数中最大的是 ( )。
- A. 296                      B. 133                      C. 256                      D. 199
12. 下列几个 32 位 IP 地址中，书写错误的是 ( )。
- A. 162.105.128.27    B. 192.168.0.1              C. 256.256.129.1    D. 10.0.0.1
13. 要求以下程序的功能是计算： $s = 1 + 1/2 + 1/3 + \dots + 1/10$ 。
- ```
#include <iostream>
using namespace std;
int main() {
    int n;
    float s;
    s = 1.0;
    for (n = 10; n > 1; n--)
        s = s + 1 / n;
    cout << s << endl;
    return 0;
}
```
- 程序运行后输出结果错误，导致错误结果的程序行是 ( )。
- A.  $s = 1.0;$                       B.  $\text{for } (n = 10; n > 1; n--)$   
C.  $s = s + 1 / n;$                       D.  $\text{cout} << s << \text{endl};$
14. 设变量  $x$  为 float 型且已赋值，则以下语句中能将  $x$  中的数值保留到小数点后两位，并将第三位四舍五入的是 ( )。
- A.  $x = (x * 100) + 0.5 / 100.0;$   
B.  $x = (x * 100 + 0.5) / 100.0;$

- C.  $x = (\text{int}) (x * 100 + 0.5) / 100.0;$   
D.  $x = (x / 100 + 0.5) * 100.0;$

15. 有以下程序:

```
#include <iostream>
using namespace std;
int main() {
    int s, a, n;
    s = 0;
    a = 1;
    cin >> n;
    do {
        s += 1;
        a -= 2;
    } while (a != n);
    cout << s << endl;
    return 0;
}
```

若要使程序的输出值为 2, 则应该从键盘给 n 输入的值是 ( )。

- A. -1                      B. -3                      C. -5                      D. 0
16. 一棵具有 5 层的满二叉树中结点数为 ( )。
- A. 31                      B. 32                      C. 33                      D. 16
17. 有向图中每个顶点的度等于该顶点的 ( )。
- A. 入度                      B. 出度  
C. 入度与出度之和                      D. 入度与出度之差
18. 设有 100 个数据元素, 采用折半搜索时, 最大比较次数为 ( )。
- A. 6                      B. 7                      C. 8                      D. 10
19. 若有如下程序段, 其中 s、a、b、c 均已定义为整型变量, 且 a、c 均已赋值,  $c > 0$ 。
- ```
s = a;
for (b = 1; b <= c; b++)
    s += 1;
```
- 则与上述程序段功能等价的赋值语句是 ( )。
- A.  $s = a + b$                       B.  $s = a + c$                       C.  $s = s + c$                       D.  $s = b + c$

20. 计算机界的最高奖是（ ）。

- A. 菲尔兹奖      B. 诺贝尔奖      C. 图灵奖      D. 普利策奖

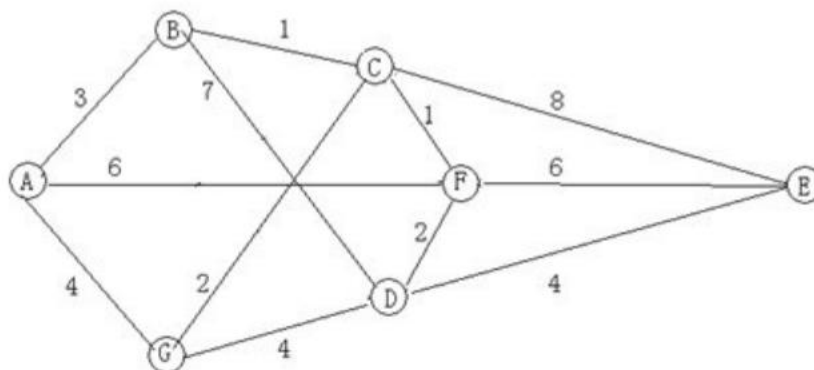
二、问题求解（共 2 题，每题 5 分，共计 10 分；每题全部答对得 5 分，没有部分分）

1. 把  $M$  个同样的球放到  $N$  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放置方法？（用  $K$  表示）。

例如： $M = 7$ ， $N = 3$  时， $K = 8$ ；在这里认为  $(5, 1, 1)$  和  $(1, 5, 1)$  是同一种放置方法。

问： $M = 8$ ， $N = 5$  时， $K =$  \_\_\_\_\_。

2. 如图所示，图中每条边上的数字表示该边的长度，则从 A 到 E 的最短距离是\_\_\_\_\_。



三、阅读程序写结果（共 4 题，每题 8 分，共计 32 分）

```
1. #include <iostream>
using namespace std;

int main() {
    int a, b, c, d, ans;
    cin >> a >> b >> c;
    d = a - b;
    a = d + c;
    ans = a * b;
    cout << "Ans = " << ans << endl;
```

```

    return 0;
}

```

输入: 2 3 4

输出: \_\_\_\_\_

2. `#include <iostream>`  
`using namespace std;`

```

int fun(int n) {
    if (n == 1)
        return 1;
    if (n == 2)
        return 2;
    return fun(n - 2) - fun(n - 1);
}

```

```

int main() {
    int n;
    cin >> n;
    cout << fun(n) << endl;
    return 0;
}

```

输入: 7

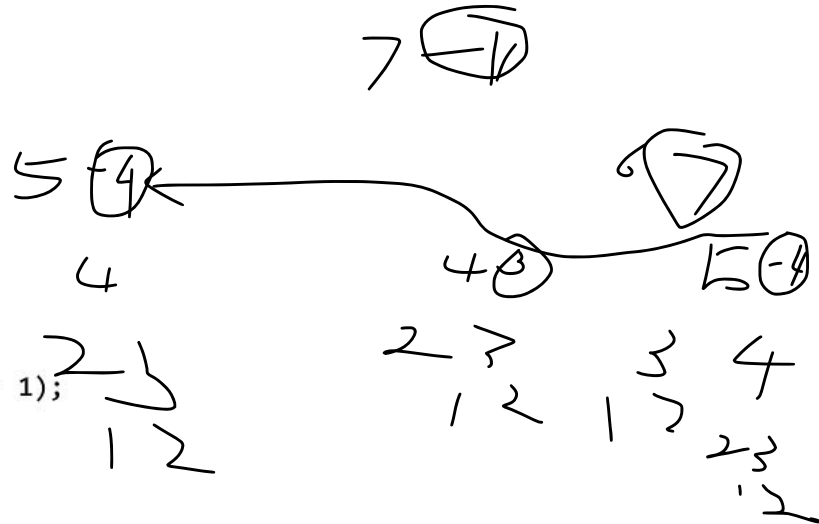
输出:   -11  

3. `#include <iostream>`  
`#include <string>`  
`using namespace std;`

```

int main()
{
    string st;
    int i, len;
    getline(cin, st);
    len = st.size();
}

```



```

    for (i = 0; i < len; i++){
        if (st[i] >= 'a' && st[i] <= 'z')
            st[i] = st[i] - 'a' + 'A';
    }
    cout << st << endl;
    return 0;
}

```

输入: Hello, my name is Lostmonkey.

输出: \_\_\_\_\_

4. #include <iostream>

using namespace std;

const int SIZE = 100;

```

int main()
{
    int p[SIZE];
    int n, tot, i, cn;
    tot = 0;
    cin >> n;
    for (i = 1; i <= n; i++)
        p[i] = 1;
    for (i = 2; i <= n; i++){
        if (p[i] == 1)
            tot++;
        cn = i * 2;
        while (cn <= n) {
            p[cn] = 0;
            cn += i;
        }
    }
    cout << tot << endl;
    return 0;
}

```

输入: 30

输出: \_\_\_\_\_

#### 四、完善程序（共 2 题，每题 14 分，共计 28 分）

1. （数字删除）下面程序的功能是将字符串中的数字字符删除后输出。请填空。（每空 3 分，共 12 分）

```
#include <iostream>
using namespace std;

int delnum(char *s) {
    int i, j;
    j = 0;
    for (i = 0; s[i] != '\0'; i++)
        if (s[i] < '0' (1) s[i] > '9') {
            s[j] = s[i];
            (2);
        }
    return (3);
}

const int SIZE = 30;

int main() {
    char s[SIZE];
    int len, i;
    cin.getline(s, sizeof(s));
    len = delnum(s);
    for (i = 0; i < len; i++)
        cout << (4);
    cout << endl;
    return 0;
}
```

2. （最大子矩阵和）给出 m 行 n 列的整数矩阵，求最大的子矩阵和（子矩阵不能为空）。

输入第一行包含两个整数  $m$  和  $n$ ，即矩阵的行数和列数。之后  $m$  行，每行  $n$  个数，描述整个矩阵。程序最终输出最大的子矩阵和。（最后一空 4 分，其余 3 分，共 16 分）

```
#include <iostream>
using namespace std;

const int SIZE = 100;
int matrix[SIZE + 1][SIZE + 1];
int rowsum[SIZE + 1][SIZE + 1]; //rowsum[i][j]记录第 i 行前 j 个数的和
int m, n, i, j, first, last, area, ans;

int main() {
    cin >> m >> n;
    for (i = 1; i <= m; i++)
        for (j = 1; j <= n; j++)
            cin >> matrix[i][j];
    ans = matrix____(1)____;
    for (i = 1; i <= m; i++)
        ____ (2) ____;
    for (i = 1; i <= m; i++)
        for (j = 1; j <= n; j++)
            rowsum[i][j] = ____ (3) ____;
    for (first = 1; first <= n; first++)
        for (last = first; last <= n; last++) {
            ____ (4) ____;
            for (i = 1; i <= m; i++) {
                area += ____ (5) ____;
                if (area > ans)
                    ans = area;
                if (area < 0)
                    area = 0;
            }
        }
    cout << ans << endl;
    return 0;
}
```