## 第二十一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: 2015年10月11日14:30~16:30

## 选手注意:

•	试题纸共有7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。
- 选

	一、单项选择题(共 20 题,每题 1.5 分, 页)	共	计 30 分;每题有且仅有一个正确
1.	1MB 等于(  )。		
	A. 1000 字节	B.	1024 字节
	C. 1000 X 1000 字节	D.	1024 X 1024 字节
2.	在 PC 机中,PENTIUM(奔腾)、酷睿、A. 生产厂家名称 B. 硬盘的型号		
3.	操作系统的作用是(  )。		
	A. 把源程序译成目标程序	В.	便于进行数据管理
	C. 控制和管理系统资源	D.	实现硬件之间的连接
,	大\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	44 ¥4	
4.	在计算机内部用来传送、存贮、加工处理 A. 二进制码 B. 八进制码		
	A	C.	近前妈 D. 省形拼音码
5.	下列说法正确的是( )。	<del>≥</del> 4.>-	4-1
	A. CPU 的主要任务是执行数据运算和程序		•
	B. 存储器具有记忆能力,其中信息任何E	, .	,
	C. 两个显示器屏幕尺寸相同,则它们的约	• // .	, , , _ , , , , , , , , , , , , , , , ,
	D. 个人用户只能使用 Wifi 的方式连接到	inte	rnet

B. 01000001 C. 01000100 D. 00111000

**6.** 二进制数 00100100 和 00010100 的和是( )。

A. 00101000

A.	0.8	В.	0.4	C.	0.2	D.	0.1
A. B.	所谓的"中断"是持操作系统随意停当出现需要时, 因停机而停止一电脑死机	止一 CPU	个程序的运行 暂时停止当前程序	字的:	执行转而执行处理	里新作	青况的过程
A. B.	人为制造的能够 一种由于计算机	的危 侵入 元器		十算	机带来故障的程序 态环境有害的物质	<b></b>	
	TP 可以用于( 远程传输文件		发送电子邮件	C.	浏览网页	D.	网上聊天
	下面哪种软件不属 <sup>-</sup> QQ		寸通信软件( ) MSN	° C.	微信	D.	P2P
<b>12.</b> 6 A.	个顶点的连通图的 6	的最才 B.		为( C.		D.	4
A. B. C.	连表不具备的特点是可随机访问任何插入、删除操作 无需事先估计存所需存储空间与	一个 不需 储空	元素 要移动元素 间大小				
A.	战性表若采用链表在必须连续 一定不连续	字储约	吉构,要求内存中	В.	存储单元地址( 部分地址必须连 连续不连续均可	续	0
•	〉有一空栈 S,对下 拄栈,进栈,出栈的		作,则此操作完成		栈 S 的栈顶元素为		) 。

7. 与二进制小数 0.1 相等的十六进制数是()。

	前序遍历序列与中	序遍	历序列相同的二叉	树为	( ) 。		
Δ	根结点无左子树	的二	. 叉树				
В	. 根结点无右子树	的二	. 叉树				
C	只有根结点的二	.叉枫	或非叶子结点只	有左	子树的二叉树		
D	. 只有根结点的二	.叉枫	或非叶子结点只	有右	子树的二叉树		
	如果根的高度为1,						
Δ	5	В.	6	C.	7	D.	8
10	下列选项中不属于	知事而-	立 <b>供</b> 枚 式 的 具 (	)			
	TXT				MOV	D.	RMVB
,	. 171	ъ.	AVI	C.	WOV	υ.	KIVIVB
19.	设某算法的计算时	间表	示为递推关系式 <b>T</b> (	n) = '	T(n - 1) + n(n 为ī	E整数	发)及 <b>T(0) = 1</b> ,则
	该算法的时间复杂			•	,		,
Δ	O(log n)	В.	O(n log n)	C.	O(n)	D.	O(n <sup>2</sup> )
20.	在 NOI 系列赛事中	参赛	选手必须使用由承	办单	位统一提供的设备	备。下	列物品中不允许选
	手自带的是()	0					
Δ	鼠标	В.	笔	C.	身份证	D.	准考证
_	问题 安留 ( 井 2	賦	<b>每颗≤分</b> #}	<u> </u>	A. 短颗 <b>人</b> 郊	¢¢ <del>√.</del> I∙	復長公 沿方郊
	问题求解(共 2	题,	每题5分,共记	<b>† 1</b> 0	分,每题全部	答对	得 5 分,没有部
二、分分		题,	每题5分,共记	<b>† 1</b> 0	分,每题全部	答对	得 5 分,没有部
分分	<del>}</del> )						
分分							
分分 1.	<b>分)</b> 重新排列 1234 使得	]每一	一个数字都不在原来	<b>长的</b> 位	立置上,一共有		
分分 1.	<del>}</del> )	]每一	一个数字都不在原来	<b>长的</b> 位	立置上,一共有		
分分 1.	<b>分)</b> 重新排列 1234 使得	]每一	一个数字都不在原来	<b>长的</b> 位	立置上,一共有		
分分 1. 2.	<b>分)</b> 重新排列 1234 使得	身每一 5 的二	一个数字都不在原来工义树最多有	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得一棵结点数为 2015 <b>阅读程序写结果</b>	身毎- ら的二 <b>・(</b> 共	一个数字都不在原来 二叉树最多有 <b>长4题,每题8</b> 分	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2.	重新排列 1234 使得 一棵结点数为 2015 <b>阅读程序写结果</b> #include <iostr< th=""><th>事毎一 。的二 :(尹 ···································</th><th>一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b></th><th><b>长的</b>位</th><th>立置上,一共有 个叶子结点。</th><th></th><th></th></iostr<>	事毎一 。的二 :(尹 ···································	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得 一棵结点数为 2015 <b>阅读程序写结果</b> #include <iostrusing namespace<="" th=""><th>事毎一 。的二 :(尹 ···································</th><th>一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b></th><th><b>长的</b>位</th><th>立置上,一共有 个叶子结点。</th><th></th><th></th></iostrusing>	事毎一 。的二 :(尹 ···································	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得一棵结点数为 2015 阅读程序写结果 #include <iostrusing namespace<br="">int main() {</iostrusing>	身每一 。 ( eam> sto	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得一棵结点数为 2015 <b>阅读程序写结果</b> #include <iostrusing namespace<br="">int main() { int a, b, c</iostrusing>	身每一 。 ( eam> sto	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得一棵结点数为 2015 阅读程序写结果 #include <iostrusing namespace<br="">int main() { int a, b, c a = 1;</iostrusing>	身每一 。 ( eam> sto	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		
分分 1. 2. 三、	重新排列 1234 使得一棵结点数为 2015 <b>阅读程序写结果</b> #include <iostrusing namespace<br="">int main() { int a, b, c</iostrusing>	身每一 。 ( eam> sto	一个数字都不在原来 工叉树最多有 <b>、 4 题,每题 8 分</b>	<b>长的</b> 位	立置上,一共有 个叶子结点。		

```
if (a > b) {
           if (a > c)
               cout << a << ' ';
           else
               cout << b << ' ';
       }
       cout << c << endl;</pre>
       return 0;
   }
   输出: _____
2. #include <iostream>
   using namespace std;
   struct point {
       int x;
       int y;
   };
   int main() {
       struct EX {
           int a;
           int b;
           point c;
       } e;
       e.a = 1;
       e.b = 2;
       e.c.x = e.a + e.b;
       e.c.y = e.a * e.b;
       cout << e.c.x << ',' << e.c.y << endl;</pre>
       return 0;
   }
   输出: ____
3. #include <iostream>
   #include <string>
   using namespace std;
```

```
int main() {
       string str;
       int i;
       int count;
       count = 0;
       getline(cin, str);
       for (i = 0; i < str.length(); i++) {</pre>
           if(str[i] >= 'a' && str[i] <= 'z')
              count++;
       }
       cout << "It has " << count << " lowercases" << endl;</pre>
       return 0;
   }
   输入: NOI2016 will be held in Mian Yang.
   输出: _____
4. #include <iostream>
   using namespace std;
   void fun(char *a, char *b) {
       a = b;
       (*a)++;
   }
   int main() {
       char c1, c2, *p1, *p2;
       c1 = 'A';
       c2 = 'a';
       p1 = &c1;
       p2 = &c2;
       fun(p1, p2);
       cout << c1 << c2 << endl;
       return 0;
   }
   输出: _____
```

## 四、完善程序(共2题,每题14分,共计28分)

**1.** (打印月历)输入月份 m (1≤m≤12),接一定格式打印 2015 年第 m 月的月历。(第三、四空 2.5 分,其余 3 分)

例如,2015年1月的月历打印效果如下(第一列为周日):

```
S
  M T W
            Т
                 F
                     S
                 2
                     3
              1
   5
          7
             8 9
       6
                     10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
#include <iostream>
using namespace std;
const int dayNum[]={-1, 31, 28, 31, 30, 31, 30, 31, 30, 31, 30, 31};
int m, offset, i;
int main() {
   cout << "S\tM\tT\tW\tT\tF\tS" << endl; // '\t'为 TAB 制表符
   (1)
   for (i = 1; i < m; i++)
       offset = (2);
   for (i = 0; i < offset; i++)
       cout << '\t';</pre>
   for (i = 1; i \leftarrow (3); i++) {
       cout << <u>(4)</u>;
       if (i == dayNum[m] || (5) == 0)
          cout << endl;</pre>
       else
          cout << '\t';
   }
   return 0;
```

}

**2. (中位数)**给定 n (n 为奇数且小于 1000) 个整数,整数的范围在  $0 \sim m$  ( $0 < m < 2^{31}$ ) 之间,请使用二分法求这 n 个整数的中位数。所谓中位数,是指将这 n 个数排序之后,排在正中间的数。(第五空 2 分,其余 3 分)

```
#include <iostream>
using namespace std;
const int MAXN = 1000;
int n, i, lbound, rbound, mid, m, count;
int x[MAXN];
int main() {
   cin >> n >> m;
   for (i = 0; i < n; i++)
       cin >> x[i];
   1bound = 0;
   rbound = m;
   while (<u>(1)</u>) {
       mid = (lbound + rbound) / 2;
       (2)
       for (i = 0; i < n; i++)
          if (<u>(3)</u>)
              (4)
       if (count > n / 2)
          lbound = mid + 1;
       else
          (5);
   }
   cout << rbound << endl;</pre>
   return 0;
}
```