

实验舱青少年信息学奥林匹克联赛初赛试题

●● 全部试题答案均要求写在答卷纸上，写在试卷纸上一律无效 ●●

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. 目前计算机芯片（集成电路）制造的主要原料是（ ），它是一种可以在沙子中提炼出的物质。
A. 硅 B. 铜 C. 锗 D. 铝
2. 地址总线的位数决定了 CPU 可直接寻址的内存空间大小，例如地址总线为 16 位，其最大的可寻址空间为 64KB。如果地址总线是 32 位，则理论上最大可寻址的内存空间为
A. 128KB B. 1MB C. 1GB D. 4GB
3. 在字符串“ababacbabcbdecced”中出现次数最多的字母出现了（ ）次
A. 5 B. 4 C. 3 D. 2
4. 以下哪个软件不是即时通信软件（ ）。
A. 网易泡泡 B. MSN Messenger C. Google Talk D. 3DS Max E. QQ
5. 下列关于高级语言的说法错误的是（ ）。
A. Fortran 是历史上的第一个面向科学计算的高级语言 B. Pascal 和 C 都是编译执行的高级语言
C. C++ 是历史上的第一个支持面向对象的语言 D. 编译器将高级语言程序转变为目标代码
E. 高级语言程序比汇编语言程序更容易从一种计算机移植到另一种计算机上
6. 下列设备不具有计算功能的是（ ）。
A. 笔记本电脑 B. 掌上电脑 C. 智能手机
D. 电子计算器 E. 液晶显示器
7. 一个字节（byte）由（ ）个二进制位组成。
A. 8 B. 16 C. 32 D. 以上皆有可能
8. 在下列 HTML 语句中，可以正确产生一个指向 NOI 官方网站的超链接的是（ ）。
A. `欢迎访问 NOI 网站`
B. `欢迎访问 NOI 网站`
C. `<a >http://www.noi.cn`
D. `欢迎访问 NOI 网站`
9. 元素 R1、R2、R3、R4、R5 入栈的顺序为 R1、R2、R3、R4、R5。如果第一个出栈的是 R3，那么第五个出栈的不可能是（ ）。

A. R1 B. R2 C. R4 D. R5

10. 常见的邮件传输服务器使用 () 协议接收邮件。

A. HTTP B. SMTP C. TCP D. FTP E. POP3

11. 一棵二叉树的前序遍历序列是 ABCDEFG, 后序遍历序列是 CBFEGDA, 则根结点的左子树的结点个数可能是 ()。

A. 2 B. 3 C. 4 D. 5

12. 在 8 位二进制补码中, 10101011 表示的数是十进制下的 ()。

A. 43 B. -85 C. -43 D. -84

13. 分辨率为 1600x900、16 位色的位图, 存储图像信息所需的空间为 ()。

2812.5KB B. 4218.75KB C. 4320KB D. 2880KB

14. 将 7 个名额分给 4 个不同的班级, 允许有的班级没有名额, 有 () 种不同的分配方案。

A. 60 B. 84 C. 96 D. 120

15. 断电后会丢失数据的存储器是 ()。

A. RAM B. ROM C. 硬盘 D. 光盘

16. 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

17. 下列几个 32 位 IP 地址中, 书写错误的是 ()。

A. 162.105.115.27 B. 192.168.0.1 C. 256.256.129.1 D. 10.0.0.1

18. 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. $x = (x / 100 + 0.5) * 100.0;$

19. 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

20. 设 A = true, B = false, C = false, D = true, 以下逻辑运算表达式值为真的是 ()

A. $(A \wedge B) \vee (C \wedge D)$ B. $((A \wedge B) \vee C) \wedge D$ C. $A \wedge ((B \vee C) \wedge D)$
D. $(A \wedge (B \vee C)) \vee D$ E. $(A \vee B) \wedge (C \wedge D)$

二. 问题求解（共 2 题，每空 5 分，共计 10 分）

1. 小陈现有 2 个任务 A, B 要完成, 每个任务分别有若干步骤如下: $A=a_1 \rightarrow a_2 \rightarrow a_3$, $B=b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4 \rightarrow b_5$. 在任何时候, 小陈只能专心做某个任务的一个步骤. 但是如果愿意, 他可以在做完手中任务的当前步骤后, 切换至另一个任务, 从上次此任务第一个未做的步骤继续. 每个任务的步骤顺序不能打乱, 例如 $\cdots a_2 \rightarrow b_2 \rightarrow a_3 \rightarrow b_3 \cdots$ 是合法的, 而 $\cdots a_2 \rightarrow b_3 \rightarrow a_3 \rightarrow b_2 \cdots$ 是不合法的. 小陈从 B 任务的 b_1 步骤开始做, 当恰做完某个任务的某个步骤后, 就停工回家吃饭了. 当他回来时, 只记得自己已经完成了整个任务 A, 其他的都忘了. 试计算小陈饭前已做的可能的任务步骤序列共有 _____ 种.

2. 有如下的一段程序:

```
a=1;
b=a;
d=-a;
e=a+d;
c=2*d;
f=b+e-d;
g=a*f+c;
```

现在要把这段程序分配到若干台（数量充足）用电缆连接的 PC 上做并行执行. 每台 PC 执行其中的某几个语句, 并可随时通过电缆与其他 PC 通讯, 交换一些中间结果. 假设每台 PC 每单位时间可以执行一个语句, 且通讯花费的时间不计. 则这段程序最快可以在 _____ 单位时间内执行完毕. 注意: 任意中间结果只有在某台 PC 上已经得到, 才可以被其他 PC 引用. 例如若语句 4 和 6 被分别分配到两台 PC 上执行, 则因为语句 6 需要引用语句 4 的计算结果, 语句 6 必须在语句 4 之后执行.

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 a,b;

int work(int a,int b){
    if (a%b)
        return work(b,a%b);
    return b;
}

int main(){
    cin >> a >> b;
    cout << work(a,b) << endl;
    return 0;
}
```

输入: 20 12
输出: _____

2.

```
#include <iostream>
using namespace std;
int main()
{
    int a[3],b[3];
    int i,j,tmp;
    for (i=0;i<3;i++)
        cin >> b[i];
    for (i=0;i<3;i++)
    {
        a[i]=0;
        for (j=0;j<=i;j++)
        {
            a[i]+=b[j];
            b[a[i]%3]+=a[j];
        }
    }
    tmp=1;
    for (i=0;i<3;i++)
    {
        a[i]%=10;
        b[i]%=10;
        tmp*=a[i]+b[i];
    }
    cout << tmp << endl;
    return 0;
}
```

输入: 2 3 5
输出: _____

3.

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

```
const int c=2009;
```

```
int main()
{
    int n,p,s,i,j,t;
    cin >> n >> p;
    s=0;t=1;
    for(i=1;i<=n;i++)
    {
        t=t*p%c;
        for(j=1;j<=i;j++)
            s=(s+t)%c;
    }
    cout << s << endl;
    return 0;
}
```

输入: 11 2

输出: _____

4.

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

const int maxn=50;
void getnext(char str[])
{
    int l=strlen(str),i,j,k,temp;
    k=l-2;
    while(k>=0&&str[k]>str[k+1]) k--;
    i=k+1;
    while(i<l&&str[i]>str[k]) i++;
    temp=str[k];
    str[k]=str[i-1];
    str[i-1]=temp;
    for(i=l-1;i>k;i--)
        for(j=k+1;j<i;j++)
            if(str[j]>str[j+1])
            {
                temp=str[j];
                str[j]=str[j+1];
                str[j+1]=temp;
            }
}
```

```

        }
    return ;
}

int main()
{
    char a[maxn];
    int n;
    cin >> a >> n;
    while(n>0)
    {
        getnext(a);
        n--;
    }
    cout << a << endl;
    return 0;
}

```

输入: NOIP 3

输出: _____

四. 完善程序 (前 8 空, 每空 3 分, 后 2 空, 每空 2 分, 共 28 分)

1. (最大连续子段和) 给出一个数列 (元素个数不多于 100), 数列元素均为负整数、正整数、0。请找出数列中的一个连续子数列, 使得这个子数列中包含的所有元素之和最大, 在和最大的前提下还要求该子数列包含的元素个数最多, 并输出这个最大和以及该连续子数列中元素的个数。例如数列为 4, -5, 3, 2, 4 时, 输出 9 和 3; 数列为 1 2 3 -5 0 7 8 时, 输出 16 和 7。

```

#include <iostream>
using namespace std;

int a[101];
int n,i,ans,len,tmp,beg;

int main(){
    cin >> n;
    for (i=1;i<=n;i++)
        cin >> a[i];
    tmp=0;
    ans=0;
    len=0;
    beg= ① ;
    for (i=1;i<=n;i++){
        if (tmp+a[i]>ans){
            ans=tmp+a[i];
            len=i-beg;
        }
    }
}

```

```

    }
    else if ( _____ ② _____ && i-beg>len)
        len=i-beg;
    if (tmp+a[i] _____ ③ _____) {
        beg= _____ ④ _____;
        tmp=0;
    }
    else
        _____ ⑤ _____;
}
cout << ans << " " << len << endl;
return 0;
}

```

2. **(国王放置)** 在 $n*m$ 的棋盘上放置 k 个国王，要求 k 个国王互相不攻击，有多少种不同的放置方法。假设国王放在第 (x,y) 格，国王的攻击的区域是： $(x-1,y-1)$ ， $(x-1,y)$ ， $(x-1,y+1)$ ， $(x,y-1)$ ， $(x,y+1)$ ， $(x+1,y-1)$ ， $(x+1,y)$ ， $(x+1,y+1)$ 。读入三个数 n,m,k ，输出答案。题目利用回溯法求解。棋盘行标号为 $0\sim n-1$ ，列标号为 $0\sim m-1$ 。

```

#include <iostream>
using namespace std;

int n,m,k,ans;
int hash[5][5];
void work(int x,int y,int tot){
    int i,j;
    if (tot==k){
        ans++;
        return;
    }
    do{
        while (hash[x][y]){
            y++;
            if (y==m){
                x++;
                y= _____ ① _____;
            }
            if (x==n)
                return;
        }
        for (i=x-1;i<=x+1;i++)
            if (i>=0&&i<n)
                for (j=y-1;j<=y+1;j++)

```

```

        if (j>=0&&j<m)
            _____②_____;
        _____③_____;
    for (i=x-1;i<=x+1;i++)
        if (i>=0&&i<n)
            for (j=y-1;j<=y+1;j++)
                if (j>=0&&j<m)
                    _____④_____;

    y++;
    if (y==m){
        x++;
        y=0;
    }
    if (x==n)
        return;
}
while (1);
}
int main(){
    cin >> n >> m >> k;
    ans=0;
    memset(hash,0,sizeof(hash));
    _____⑤_____;
    cout << ans << endl;
    return 0;
}

```