



初赛模赛

选择题

快快编程
kkcoding.net

选择

1. 下列哪个选项（ ）可以正确用来表示八进制数。

A、0x18 B、(0800)₈ C、123O

D、(001)₂

答案：C

解释：O作为结尾表示八进制，0x在前面表示16进制

选择

2. 计算机各个部件传输信息的公共通路被称为总线，一次传输信息的位数称为总线的（ ）。

A、长度 B、粒度 C、宽度 D、深度

答案：C

解释：计算机总线一次传输信息的位数称为宽度

选择

3. 先于或随着操作系统的系统文件装入内存储器，从而获得计算机特定控制权并进行串扰和破坏的病毒是（ ）。

- A、文件型病毒
- B、引导区型病毒
- C、宏病毒
- D、网络病毒

答案：B

解释：引导型病毒指寄生在次盘引导区或主引导区的计算机病毒。此种病毒利用系统引导时，不对主引导区的内容正确与否进行判别的缺点，在引导系统的过程中侵入系统、驻留内存、监视系统运行、待机传染和破坏

选择

4. 在平衡二叉树中，节点的平衡因子的取值不可能是（ ）。

A、1 B、2 C、-1 D、0

答案：B

解释：平衡二叉树任意左右子树的高度差值绝对值不超过1，则平衡二叉树节点的平衡因子的值只可能是-1，0，1

选择

5. 六个人排成一排，甲乙不相邻，且甲丙不相邻的不同排法数是多少种（ ）。

- A. 216
- B. 240
- C. 288
- D. 360

答案：C

解释：首先将甲乙丙拿出来，剩下三个做全排列，有 $A(3,3)=6$ 种排列，然后将甲乙两个人插入第一步三个人的四个空隙当中，有 $A(4,2)=12$ 种，剩下丙插入前五个人的六个空隙，排除掉甲旁边的两个空隙，还有 $C(4,1)=4$ ，总共有 $6*12*4=288$ 种

选择

6. 已有一个栈，进栈顺序为**ABCD**，出栈顺序为**BCDA**，若用**I**表示进栈，**O**表示出栈，则进操作的顺序是（ ）。

- A. IOIOIIIO
- B. IIOOIIIO
- C. IIOIOIOO
- D. IIOIOOOIO

答案：C

解释：根据出栈顺序，模拟，先**AB**进，**B**出，**C**进，**C**出，**D**进，**D**出，**A**出

选择

7. 十六进制数FFF.CH相当于十进制数（ ）。

- A. 4096.3
- B. 4096.25
- C. 4095.75
- D. 4096.75

答案： C

解释： 进制转换

选择

8. 面向对象程序设计方法的优点不包括（ ）。

A. 可重用性 B. 可扩展性 C. 易于管理和维护 D. 简单易懂

答案：D

解释：面向对象涉及到一定思想，没有那么简单，只是相对面向过程思想上有所优化

选择

9. 在一个图中，所有顶点的度数之和等于图的边数的（ ）倍。
A. 2 B. 1 C. $1/2$ D. 4

答案：A

解释：一条边对应一个源节点和一个终结点，任意一条边都对应2个度，所以度数总和是边数总和的两倍

选择

10. 线性表(a1,a2,...,an)以链接方式存储时，访问第i位置元素的时间复杂性为（ ）。

- A. $O(i)$
- B. $O(1)$
- C. $O(n)$
- D. $O(i-1)$

答案：C

解释：访问第一个的时间为1，第二个的时间为2，以此类推，访问任意一个的平均时间为 $(1+n)*n/2n$ ，时间复杂度为 $O(n)$

选择

11. 4个袋子，15个球，每个袋子至少放一个球，而且袋子中的球的数量不能重复，问有多少种方式（ ）。

- A. 4
- B. 6
- C. 7
- D. 5

答案：B

解释：数量比较少，直接枚举

选择

12.3*(5-2)+7的后缀表达式为（ ）。

A. 352-*7+

B. 532-*7+

C. 352*-7+

D. 352-*+7

答案：A

解释：中缀表达式转换后缀表达式

选择

13. 下列标识符错误的事（ ）。

- A. xly
- B. _123
- C. 2ab
- D. _ab

答案： C

解释： 标识符必须以字母**a-z**、**A-Z**活着下划线开头

选择

14. 若是一棵完全二叉树有768个节点，则该二叉树中叶节点的个数为（ ）。

A. 257 B. 258 C. 384 D. 385

答案：C

解释：对于完全二叉树而言，当节点数 n 是奇数，无度为1的节点； n 为偶数时，有一个度为1的节点， $n_0 = n_2 + 1$ ，题中节点数为偶数，所以有一个度为1的节点。减去1个度为1的节点之后，还有767个，此时 $n_0 + n_2 = 767$ ， $n_0 = 384$

选择

15. 对于一个具有 n 个顶点的无向图，若是采用邻接表表示，则存放表头节点的数组大小为（ ）。

- A. n
- B. $n+1$
- C. $n-1$
- D. n +边数

答案：A

解释：邻接表的性质决定，存在多少个节点，就有多少个头节点的数组，每个头节点的数组都指向该节点在图中直接相连的节点

阅读程序

快快编程
kkcoding.net

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

判断

1. 本程序时间复杂度为 $O(n)$ 。()

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

判断

2. maxx的值必然不可能大于n。()

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

判断

3. maxx的初始值没有给出。()

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

判断

4. 12行s>maxx修改为s>=maxx会导致程序结果受到影响。()

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

选择	<p>5. 若输入 10 3 5 6 2 3 4 5 6 8 9 程序输出结果为 ()。</p> <p>A. 1 B. 6 C. 7 D. 5</p>
----	--

阅读程序

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n,q,i,j,s=1,maxx;
4  int main()
5  {
6      cin>>n>>q;
7      for(i=2;i<=n;i++)
8      {
9          cin>>j;
10         if(q<j&& j-q==1) s++;
11         else s=1;
12         if(s>maxx) maxx=s;
13         q=j;
14     }
15     cout<<maxx;
16     return 0;
17 }
```

选择

6. 若输一串质数，理论上最大的输出结果为（ ）
A. 1 B. 2 C. 0 D. 最大值不确定

阅读程序

快快编程
kkcoding.net

2分钟

阅读程序

```
1  #include<stdio>
2  #include<queue>
3  using namespace std;
4
5  queue<char>a;
6  int left,right,t;
7
8  void read(){
9      char ch=getchar();
10     while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t)--t;
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

- 1.请提取所有变量
- 2.选择题中手算样例
3. 结合自己的经验，分析

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t--t;
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

选择

5. 若输入

$2*(x+y)/(1-x)@$

输出结果是()。

A. YES B. NO C. YES! D. NO!

手算样例

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==''){
21             ++right;
22             if(t--t;
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

选择

6. 若输入

$(25+x)*(a*(a+b+b))@$

输出结果时 ()

A. YES B. NO C. YES! D. NO!

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t-->0);
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

判断

1. 程序输入直到读到'@'为止。()

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t--t;
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

判断

2. t代表读入的左括号数量。()

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t-->0);
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

判断

3. 26行比较条件仅仅`left==right`程序输出结果不变。()

2分钟

阅读程序

```
1 #include<stdio>
2 #include<queue>
3 using namespace std;
4
5 queue<char>a;
6 int left,right,t;
7
8 void read(){
9     char ch=getchar();
10    while(ch!='@') a.push(ch),ch=getchar();
11 }
12
13 int main(){
14     read();
15     while(!a.empty()){
16         if(a.front()=='('){
17             ++left;
18             ++t;
19         }
20         if(a.front()==')'){
21             ++right;
22             if(t-->0);
23         }
24         a.pop();
25     }
26     if(left==right&& t==0) printf("YES");
27     else printf("NO");
28     return 0;
29 }
```

判断

4. 输入 (())) ((, 26行做判定时left和right不相等。()

阅读程序

快快编程
kkcoding.net

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include<cstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

- 1.请提取所有变量
- 2.选择题中手算样例
- 3.结合自己的经验，分析

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include&ltcstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

选择

3. 输入

BADC

BDCA

输出是 ()。

A. ABCD

B. ABDC

C. ACBD

D. ACDB

手算样例

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include&ltcstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

选择

4. 输入

BADC

BDCA

会执行几次第10行的beford
函数（ ）次。

A. 4

B. 3

C. 2

D. 1

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include<cstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

选择

5. 长度为 n 的序列，最差的情况，会迭代多少次beford函数（ ）

- A. n
- B. $n \log n$
- C. $n+1$
- D. $n/2$

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include&ltcstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

选择

6. 输入

342165897

432698751

输出是（ ）。

A. 123456789

B. 124357689

C. 123456879

D. 156789234

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include<cstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

判断

1. 程序蕴藏了递归的思想。
()

3分钟

阅读程序

```
1 #include<cstdio>
2 #include<iostream>
3 #include<cstring>
4 using namespace std;
5 void beford(string in,string after){
6     if (in.size()>0){
7         char ch=after[after.size()-1];
8         cout<<ch;
9         int k=in.find(ch);
10        beford(in.substr(0,k),after.substr(0,k));
11        beford(in.substr(k+1),after.substr(k,in.size()-k-1));
12    }
13 }
14 int main(){
15     string inord,aftord;
16     cin>>inord;
17     cin>>aftord;
18     beford(inord,aftord);cout<<endl;
19     return 0;
20 }
```

判断

2. cout<<ch一共要运行
inord.size()次。()

完善程序

快快编程
kkcoding.net

完善程序

1. (阶乘之和) 用高精度计算出

$$S=1!+2!+3!+4!+\dots+n!(n\leq 50)$$

其中! 代表阶乘, 例如 $5!=5*4*3*2*1$

输入:

正整数N。

输出:

一个正整数S, 表示计算结果。

样例:

输入:

3

输出

9

完善程序

3分钟

```
1 #include<iostream>
2 #include<cstring>
3 using namespace std;
4 int n,a[90],b[90],c[90],f[90],d=0,len_a,len_b=1,len_c=1,len_ans,m=1;
5 string s;
6 int main(){
7     cin>>n;
8     b[0]=1;
9     for(int i=1;i<=n;i++){
10         len_a=0;
11         int p=i;
12         while(p>0){
13             a[len_a++]=p%10;
14             ---1---;
15         }
16         for(int j=0;j<len_a;j++)
17             for(int k=0;k<=len_b;k++)
18                 c[---2---]+=a[j]*b[k];
19         for(int j=0;j<len_c;j++)
20             if(c[j]>9) c[j+1]+=c[j]/10,c[j]%=10;
21         if(c[len_c]) len_c++;
22         len_ans=len_b,len_b=len_c,m=max(m,len_c);
23         for(int k=len_c-1;k>=0;k--) ---3---;
24         len_c=len_a+len_ans;
25         memset(c,0,sizeof(c));
26         for(int j=0;j<m;j++){
27             ---4---;
28             if(f[j]>9) f[j+1]+=f[j]/10,f[j]%=10;
29         }
30     }
31     while(!f[m]&& m>0) ---5---;
32     for(int i=m;i>=0;i--) cout<<f[i];
33     return 0;
34 }
```

- 1.程序太长就先提取关键部分
- 2.手算样例跟着程序走
- 3.理解题目中提供的思路

切记一点，填空已经变成选择题，结合题目！！

完善程序

```
for(int i=1;i<=n;i++){
    len_a=0;
    int p=i;
    while(p>0){
        a[len_a++]=p%10;
        ---1---;
    }
    for(int j=0;j<len_a;j++)
        for(int k=0;k<=len_b;k++)
            c[---2---]+=a[j]*b[k];
    for(int j=0;j<len_c;j++)
        if(c[j]>9) c[j+1]+=c[j]/10,c[j]%10;
    if(c[len_c]) len_c++;
    len_ans=len_b,len_b=len_c,m=max(m,len_c);
    for(int k=len_c-1;k>=0;k--) ---3---;
    len_c=len_a+len_ans;
    memset(c,0,sizeof(c));
    for(int j=0;j<m;j++){
        ---4---;
        if(f[j]>9) f[j+1]+=f[j]/10,f[j]%10;
    }
}
```

选择

1. (1) 处应填 ()。
- A. p/=10 B. p-- C. p-=10 D. p++

完善程序

```
for(int i=1;i<=n;i++){
    len_a=0;
    int p=i;
    while(p>0){
        a[len_a++]=p%10;
        ---1---;
    }
    for(int j=0;j<len_a;j++)
        for(int k=0;k<=len_b;k++)
            c[---2---]+=a[j]*b[k];
    for(int j=0;j<len_c;j++)
        if(c[j]>9) c[j+1]+=c[j]/10,c[j]%10;
    if(c[len_c]) len_c++;
    len_ans=len_b,len_b=len_c,m=max(m,len_c);
    for(int k=len_c-1;k>=0;k--) ---3---;
    len_c=len_a+len_ans;
    memset(c,0,sizeof(c));
    for(int j=0;j<m;j++){
        ---4---;
        if(f[j]>9) f[j+1]+=f[j]/10,f[j]%10;
    }
}
```

选择

2. (2) 处应填 ()。

A. j+k-1 B. j+k C. k D. k+1

完善程序

```
for(int i=1;i<=n;i++){
    len_a=0;
    int p=i;
    while(p>0){
        a[len_a++]=p%10;
        ---1---;
    }
    for(int j=0;j<len_a;j++)
        for(int k=0;k<=len_b;k++)
            c[---2---]+=a[j]*b[k];
    for(int j=0;j<len_c;j++)
        if(c[j]>9) c[j+1]+=c[j]/10,c[j]%10;
    if(c[len_c]) len_c++;
    len_ans=len_b,len_b=len_c,m=max(m,len_c);
    for(int k=len_c-1;k>=0;k--) ---3---;
    len_c=len_a+len_ans;
    memset(c,0,sizeof(c));
    for(int j=0;j<m;j++){
        ---4---;
        if(f[j]>9) f[j+1]+=f[j]/10,f[j]%10;
    }
}
```

选择

3. (3) 处应填 ()。

- A. $c[k]=b[k]$ B. $c[k]--$ C. $a[k]=c[k]$
D. $b[k]=c[k]$

完善程序

```
for(int i=1;i<=n;i++){
    len_a=0;
    int p=i;
    while(p>0){
        a[len_a++]=p%10;
        ---1---;
    }
    for(int j=0;j<len_a;j++)
        for(int k=0;k<=len_b;k++)
            c[---2---]+=a[j]*b[k];
    for(int j=0;j<len_c;j++)
        if(c[j]>9) c[j+1]+=c[j]/10,c[j]%10;
    if(c[len_c]) len_c++;
    len_ans=len_b,len_b=len_c,m=max(m,len_c);
    for(int k=len_c-1;k>=0;k--) ---3---;
    len_c=len_a+len_ans;
    memset(c,0,sizeof(c));
    for(int j=0;j<m;j++){
        ---4---;
        if(f[j]>9) f[j+1]+=f[j]/10,f[j]%10;
    }
}
```

选择

4. (4) 处应填 ()。

A. $f[j] += b[j] + a[j]$ B. $f[j] += a[j]$ C. $f[j] += c[j]$ D. $f[j] += b[j]$

完善程序

```
for(int i=1;i<=n;i++){
    len_a=0;
    int p=i;
    while(p>0){
        a[len_a++]=p%10;
        ---1---;
    }
    for(int j=0;j<len_a;j++)
        for(int k=0;k<=len_b;k++)
            c[---2---]+=a[j]*b[k];
    for(int j=0;j<len_c;j++)
        if(c[j]>9) c[j+1]+=c[j]/10,c[j]%=10;
    if(c[len_c]) len_c++;
    len_ans=len_b,len_b=len_c,m=max(m,len_c);
    for(int k=len_c-1;k>=0;k--) ---3---;
    len_c=len_a+len_ans;
    memset(c,0,sizeof(c));
    for(int j=0;j<m;j++){
        ---4---;
        if(f[j]>9) f[j+1]+=f[j]/10,f[j]%=10;
    }
}
while(!f[m]&& m>0) ---5---;
for(int i=m;i>=0;i--) cout<<f[i];
return 0;
```

选择

5. (5) 处应填 ()。

- A. m-=2 B. m-- C. f[m]--
D. f[m--]++

完善程序

快快编程
kkcoding.net

完善程序

2.（乒乓球）国际乒联现在主席沙拉拉自从上任以来就立志于推行一系列改革，以推动乒乓球运动在全球的普及。其中11分制改革引起了很大的争议，有一部分球员因为无法适应新规则只能选择退役。华华就是其中一位，他退役之后走上了乒乓球研究工作，意图弄明白

11分制和21分制对选手的不同影响。在开展他的研究之前，他首先需要对他多年比赛的统计数据进行一些分析，所以需要你的帮忙。

华华通过以下方式进行分析，首先将比赛每个球的胜负列成一张表，然后分别计算在

11分制和21分制下，双方的比赛结果（截至记录末尾）。

比如现在有这么一份记录，（其中W表示华华获得一分，L表示华华对手获得一分）：

WWLW

在11分制下，此时比赛的结果是华华第一局11比0获胜，第二局11比0获胜，正在进行第三局，当前比分1比1。而在21分制下，此时比赛结果是华华第一局21比0获胜，正在进行第二局，比分

2比1。如果一局比赛刚开始，则此时比分为0比0。直到分差大于或者等于2，才一局结束。

你的程序就是要对于一系列比赛信息的输入（WL形式），输出正确的结果。

手算样例

输入格式

每个输入文件包含若干行字符串，字符串有大写的W、L和E组成。其中E表示比赛信息结束，程序应该忽略E之后的所有内容。

输出格式

输出由两部分组成，每部分有若干行，每一行对应一局比赛的比分（按比赛信息输入顺序）。其中第一部分是11分制下的结果，第二部分是21分制下的结果，两部分之间由一个空行分隔。

样例

输入：

WWLWE

输出:

11:0

11:0

1:1

21:0

2:1

完善程序

```
char s;  
for(int i=1;cin>>s&&---1---;i++)  
{  
    if(s=='W') win[i]=1;  
    else win[i]=2;  
}
```

- 1.程序基本组成部分：输入，输出
- 2.带上手算样例，边看边写下遇到的变量
- 3.分析要结合选择题选项

完善程序

```
char s;  
for(int i=1;cin>>s&&---1---;i++)  
{  
    if(s=='W') win[i]=1;  
    else win[i]=2;  
}
```

选择

1. (1) 处应填 ()。

A. s!='E' B. s!= 'W' C. s!= 'L' D. s!= ' '

完善程序

```
for(int i=1;1;i++)
{
    if(win[i]==1) ---2---;
    if(win[i]==2) l++;
    if(win[i]==0)
    {
        cout<<w<<":"<<l<<endl<<endl;
        ---3---;
    }
    if(w-l>=2||l-w>=2)
        if(---4---)
        {
            cout<<w<<":"<<l<<endl;
            w=0;
            l=0;
        }
}
```

选择

2. (2) 处应填 () 。

A. l-- B. w++

C. w+=2 D. l-=2

完善程序

```
for(int i=1;1;i++)
{
    if(win[i]==1) ---2---;
    if(win[i]==2) l++;
    if(win[i]==0)
    {
        cout<<w<<":"<<l<<endl<<endl;
        ---3---;
    }
    if(w-l>=2||l-w>=2)
        if(---4---)
        {
            cout<<w<<":"<<l<<endl;
            w=0;
            l=0;
        }
}
```

选择

3. (3) 处应填 ()。
A. continue B. break C.
return 0 D. cout<<endl

完善程序

```
for(int i=1; i<=10; i++)
{
    if(win[i]==1) ---2---;
    if(win[i]==2) l++;
    if(win[i]==0)
    {
        cout<<w<<": "<<l<<endl<<endl;
        ---3---;
    }
    if(w-l>=2||l-w>=2)
        if(---4---)
        {
            cout<<w<<": "<<l<<endl;
            w=0;
            l=0;
        }
}
```

选择

4. (4) 处应填 ()。

A. $w \geq 11$

B. $l \geq 11$

C. $w \geq 11 \parallel l \geq 11$

D. $w \geq 11 \&\& l \geq 11$

完善程序

```
w=0;
l=0;
for(int i=1;i++)
{
    if(win[i]==1)w++;
    if(win[i]==2)l++;
    if(win[i]==0)
    {
        cout<<w<<": "<<l;
        break;
    }
    if(---5---)
        if(w>=21||l>=21)
        {
            cout<<w<<": "<<l<<endl;
            w=0;
            l=0;
        }
}
```

选择

5. (5) 处应填 ()。

A. $w-l \geq 2 \&\& l-w \geq 2$

B. $w-l \geq 2 || l-w \geq 2$

C. $w-l \geq 1 \&\& l-w \geq 1$

D. $w-l \geq 1 || l-w \geq 1$