

# 专题训练

# 阅读程序

快快编程  
kkcoding.net

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin, s1);
7      getline(cin, s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

1

识别变量

常见变量名  
翻译循环变量  
根据变量名的英文推断

2

找出关键语句

控制结构(for, if)  
常见算法的基本操作  
函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 阅读程序

## 解释变量的作用

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

s1

输入第一行的字符串

s2

输入第二行的字符串

length1

s1的长度

length2

s2的长度

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin, s1);
7      getline(cin, s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

关键词句

遍历字符串s1

将小写字符s1[i]转为大写

遍历字符串s2

将小写字符s2[i]转为大写

比较s1和s2的字典序

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

判断

将第10行修改为for (i = 1; i <= length1; i++), 程序运行会出错。( )

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

判断 将第18行开头的else删除，程序运行结果不变。( )



# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

判断

将第6行改为cin>>s1，第7行改为cin>>s2，程序运行结果不变。  
( )



# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

判断 将11行、14行删除，程序运行结果不变( )

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

选择

输入第一行 “I have a dream.”，第12行共执行（ ）次。

A. 8    B. 9    C. 10    D. 11

# 阅读程序

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int i, length1, length2;
5      string s1, s2;
6      getline(cin,s1);
7      getline(cin,s2);
8      length1 = s1.size();
9      length2 = s2.size();
10     for (i = 0; i < length1; i++)
11         if (s1[i] >= 'a' && s1[i] <= 'z')
12             s1[i] -= 'a' - 'A';
13     for (i = 0; i < length2; i++)
14         if (s2[i] >= 'a' && s2[i] <= 'z')
15             s2[i] -= 'a' - 'A';
16     if (s1 == s2)
17         cout << "=" << endl;
18     else if (s1 > s2)
19         cout << ">" << endl;
20     else
21         cout << "<" << endl;
22     return 0;
23 }
```

选择

输入第一行 “I have a dream.”，输入第二行” I Have A Dream.”，程序的输出是( )。

A. =    B. >    C. <    D. 无输出

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

1

识别变量

常见变量名  
翻译循环变量  
根据变量名的英文推断

2

找出关键语句

控制结构(for, if)  
常见算法的基本操作  
函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 阅读程序

## 解释变量的作用

st

输入字符串

## 关键语句

遍历字符串st

将大写字符st[i]改写  
为st[i]+1

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

判断 输入的字符只能由大写字母或小写字母组成 ( ) 。

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

判断 程序第5行++i改写成i++，程序运行结果不变（ ）。



# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

判断 程序的输入和输出一定不同（ ）。

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

判断 输入全部是小写，输出结果与输入结果相同（ ）。

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

选择 输入QuanGuoLianSai，输出是（ ）。

- A. QuanGuoLianSai    B. quanguoliainsai  
C. RuanHuoMianTai    D. QvboGvpLjboSbj

# 阅读程序

```
1  #include<stdio>
2  char st[100];
3  int main() {
4      scanf("%s", st);
5      for (int i = 0; st[i]; ++i) {
6          if ('A' <= st[i] && st[i] <= 'Z')
7              st[i] += 1;
8      }
9      printf("%s\n", st);
10     return 0;
11 }
```

选择 输入Z，输出是（ ）

A. Z

B. A

C. z

D. [

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

1

识别变量

常见变量名  
翻译循环变量  
根据变量名的英文推断

2

找出关键语句

控制结构(for, if)  
常见算法的基本操作  
函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 阅读程序

## 解释变量的作用

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

t[]

计数器数组

s

输入的字符串

t[i]

ASCII值为i的字符出现次数

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

怎样理解字符作为  
数组下标

字符类型的ASCII  
代码数值0-127

将字符的数值作为  
下标



# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

关键语句

计数器数组清零

统计字符s[i]出现的次数

按照字符串的顺序输出第一个出现一次的字符

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

判断

8、9两行代码在程序中没有实际作用，可以删除，不影响运行结果( )

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

判断 如果输入的字符串长度超过256，第11行会数组越界( )

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

判断 当输入的字符串只包含数字时，程序总是输出no( )

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

判断 无论输入，输出要么只有1个字符，要么输出no ( )

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

选择 输入yazxaz，程序输出结果( )。

A. a    B. x    C. y    D. z

# 阅读程序

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int t[256];
5      string s;
6      int i;
7      cin>>s;
8      for (i = 0; i < 256; i++)
9          t[i] = 0;
10     for(i = 0; i < s.length(); i++)
11         t[s[i]]++;
12     for (i = 0; i < s.length(); i++)
13         if (t[s[i]]== 1) {
14             cout << s[i] << endl;
15             return 0;
16         }
17     cout<< "no" << endl;
18     return 0;
19 }
```

选择 下面哪个输入，程序输出no（ ）

A. abcd bca

B. http

C. www

D. happy



完善程序



# 求字符串的逆序

下面的程序的功能是输入若干行字符串，每输入一行，就按逆序输出该行，最后键入-1 终止程序。请将程序补充完整。

# 手算样例

输入

CDEF  
DFVC  
-1

输出

FEDC  
CVFD

输入

QWERTYUIOP  
ASDFGHJKL  
-1  
ZXCVBNM

输出

POIUYTREWQ  
LKJHGFDSA

# 完善程序

```
1  #include <iostream>
2  #include <cstring>
3  using namespace std;
4  int maxline=200,kz;
5  int reverse(char s[])
6  {
7      int i,j,t;
8      for(i=0,j=strlen(s)-1; i<j;--1--,--2--)
9      {
10         t=s[i]; s[i]=s[j]; s[j]=t;
11     }
12     return 0;
13 }
14 int main(){
15     char line[100];
16     cout<<"continue? -1 for end."<<endl;
17     cin>>kz;
18     while(--3--)
19     {
20         cin>>line;
21         --4--;
22         cout<<line<<endl;
23         cout<<"continue? -1 for end."<<endl;
24         cin>>kz;
25     }
26     return 0;
27 }
```

1

识别变量

常见变量名  
翻译循环变量  
根据变量名的英文推断

2

找出关键语句

控制结构(for, if)  
常见算法的基本操作  
函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 完善程序

```
1  #include <iostream>
2  #include <cstring>
3  using namespace std;
4  int maxline=200,kz;
5  int reverse(char s[])
6  {
7      int i,j,t;
8      for(i=0,j=strlen(s)-1; i<j;--1--,--2--)
9      {
10         t=s[i]; s[i]=s[j]; s[j]=t;
11     }
12     return 0;
13 }
14 int main(){
15     char line[100];
16     cout<<"continue? -1 for end."<<endl;
17     cin>>kz;
18     while(--3--)
19     {
20         cin>>line;
21         --4--;
22         cout<<line<<endl;
23         cout<<"continue? -1 for end."<<endl;
24         cin>>kz;
25     }
26     return 0;
27 }
```

strlen()	求字符串长度
string 字符串	length 长度

kz	控制变量，程序是否继续 循环
----	-------------------

reverse()	反转，将字符串反转
-----------	-----------

i	左侧字符下标
---	--------

j	右侧字符下标
---	--------

line	输入一行字符串
------	---------

# 完善程序

## 关键语句

从字符串两端依次交换编号*i,j*的字符,反转字符串*s*

交换变量*s[i]*和*s[j]*

输入-1结束程序  
读入一个字符串并反转

```
1  #include <iostream>
2  #include <cstring>
3  using namespace std;
4  int maxline=200,kz;
5  int reverse(char s[])
6  {
7      int i,j,t;
8      for(i=0,j=strlen(s)-1; i<j;--1--,--2--)
9      {
10         t=s[i]; s[i]=s[j]; s[j]=t;
11     }
12     return 0;
13 }
14 int main(){
15     char line[100];
16     cout<<"continue? -1 for end."<<endl;
17     cin>>kz;
18     while(--3--)
19     {
20         cin>>line;
21         --4--;
22         cout<<line<<endl;
23         cout<<"continue? -1 for end."<<endl;
24         cin>>kz;
25     }
26     return 0;
27 }
```

# 完善程序

(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。

快快编程  
kkcoding.net



# 手算样例

输入样例：

1He2llo3 World 456!

输出多少？

输出样例：

Hello World!

快快编程  
kkcoding.net

5分钟

# 完善程序

```
1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____ (1) _____ s[i] > '9') {
9              s[j] = s[i];
10             _____ (2) _____;
11         }
12     return _____ (3) _____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____ (4) _____;
21     cout << endl;
22     return 0;
23 }
```

1

识别变量

常见变量名

翻译循环变量

根据变量名的英文推断

2

找出关键词句

控制结构(for, if)

常见算法的基本操作

函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 完善程序

```
1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____ (1) _____ s[i] > '9'){
9              s[j] = s[i];
10             _____ (2) _____;
11         }
12     return _____ (3) _____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____ (4) _____;
21     cout << endl;
22     return 0;
23 }
```

delnum函数  
输入一个字符串  
输出一个整数

C语言字符串以'\0'特殊字符表示字符串末尾

s表示输入的字符数组

len表示去掉数字字符后  
所剩字符数组的长度

# 完善程序

```

1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____(1)_____ s[i] > '9'){
9              s[j] = s[i];
10             _____(2)_____;
11         }
12     return _____(3)_____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____(4)_____;
21     cout << endl;
22     return 0;
23 }

```

cin.getline函数  
通过cin将一个字符串  
输入到字符数组s中

第二个参数规定了最多可  
允许读入 SIZE-1个字符

# 完善程序

```

1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____(1)_____ s[i] > '9'){
9              s[j] = s[i];
10             _____(2)_____;
11         }
12     return _____(3)_____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____(4)_____;
21     cout << endl;
22     return 0;
23 }

```

通过调用delnum函数  
返回删除数字字符后所剩字符数  
然后根据字符数进行循环  
重新输出改动后的字符数组

# 完善程序

```

1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____ (1) _____ s[i] > '9'){
9              s[j] = s[i];
10             _____ (2) _____;
11         }
12     return _____ (3) _____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____ (4) _____;
21     cout << endl;
22     return 0;
23 }

```

对s中每个字符i  
如果s[i]不是**数字字符**  
就把该字符赋值到字串前部  
也就是s[j]=s[i] (j从0开始)

# 完善程序

```
1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____ (1) _____ s[i] > '9') {
9              s[j] = s[i];
10             _____ (2) _____;
11         }
12     return _____ (3) _____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____ (4) _____;
21     cout << endl;
22     return 0;
23 }
```

传入字符数组s  
对s中每个字符i  
如果s[i]不是数字字符  
就把该字符赋到原字符串前部  
结束后返回新字符串长度j

定义变量并cin输入字符串s

返回得到新字符串长度  
并根据长度控制输出新字符串



# 完善程序

```
1  #include <iostream>
2  using namespace std;
3  const int SIZE = 30;
4  int delnum(char *s) {
5      int i, j;
6      j = 0;
7      for (i = 0; s[i] != '\0'; i++)
8          if (s[i] < '0' _____ (1) _____ s[i] > '9') {
9              s[j] = s[i];
10             _____ (2) _____;
11         }
12     return _____ (3) _____;
13 }
14 int main() {
15     char s[SIZE];
16     int len, i;
17     cin.getline(s, sizeof(s));
18     len = delnum(s);
19     for (i = 0; i < len; i++)
20         cout << _____ (4) _____;
21     cout << endl;
22     return 0;
23 }
```

根据ASCII码表规律  
判断不是数字字符的条件应是  
 $s[i] < '0'$  或者  $s[i] > '9'$

$s[i]$ 赋值给 $s[j]$ 后，编号 $j$ 往后移动

18行用整型 $len$ 接收 $delnum$ 返回值  
结合19行可知，返回值是原字符串中去  
掉数字字符后的字符总数，也就是 $j$

题目要求  
重新输出去掉数字字符后的新字符串  
所以用 $for$ 依次输出其中每个元素 $s[i]$



# 完善程序

**（字符串替换）** 给定一个字符串**S**（**S**仅包含大小写字母），下面的程序将**S**中的每个字母用规定的字母替换，并输出**S**经过替换后的结果。程序的输入是两个字符串，第一个字符串是给定的字符串**S**，第二个字符串**S'**由26个字母组成，它是a-z的任一排列，大小写不定，**S'**规定了每个字母对应的替换字母：**S'**中的第一个字母是字母**A**和**a**的替换字母，即**S**中的**A**用该字母的大写替换，**S**中的**a**用该字母的小写替换；**S'**中的第二个字母是字母**B**和**b**的替换字母，即**S**中的**B**用该字母的大写替换，**S**中的**b**用该字母的小写替换；..... 以此类推。

# 手算样例

输入

ABCD

BCDEFGHIJKLMNOPQRSTUVWXYZA

输出

BCDE

输入

TYHGHKN

QWERTYUIOPasdfghjklzxcvbnm

输出

ZNIUIAF

快快编程  
kkcoding.net

# 手算样例

原文	A	B	C	D	E	F	G	H	I	G	K	L	M
密文	Q	W	E	R	T	Y	U	I	O	P	A	S	D
原文	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
密文	F	G	H	J	K	L	Z	X	C	V	B	N	M

输入

SOeasy  
QWERTYUIOPasdfghjklzxcvbnm

输出

L	G	t	q	l	n
---	---	---	---	---	---

快快编程  
kkcoding.net

# 完善程序

```
1  #include <iostream>
2  #include <cstring>
3  char change[26], str[5000];
4  using namespace std;
5  void CheckChangeRule()
6  {
7      int i;
8      for (i = 0; i < 26; i++)
9      {
10         if ( _____1_____ )
11             change[i] -= 'A' - 'a';
12     }
13 }
14 void ChangeString()
15 {
16     int i;
17     for (i = 0; i < strlen(str); i++)
18     {
19         if ( _____2_____ )
20             str[i] = change[str[i] - 'A' - 'a' + 'A'];
21         else
22             _____3_____
23     }
24 }
25 int main()
26 {
27     int i;
28     cin >> str ;
29     cin >> change;
30     CheckChangeRule();
31     _____4_____
32     cout << str << endl;
33     return 0;
34 }
```

1

识别变量

常见变量名  
翻译循环变量  
根据变量名的英文推断

2

找出关键词句

控制结构(for, if)  
常见算法的基本操作  
函数参数、返回值

3

理解代码段作用

翻译解释代码段

# 完善程序

```

1  #include <iostream>
2  #include <cstring>
3  char change[26], str[5000];
4  using namespace std;
5  void CheckChangeRule()

```

change[]

26个元素，存储每个字符的替换规则

str[]

输入的需要加密的字符串

CheckChangeRule()

检查替换规则，统一替换规则，大小写保持一致

```

6  {
7      int i;
8      for (i = 0; i < 26; i++)
9      {
10         if ( _____1_____ )
11             change[i] -= 'A' - 'a';
12     }
13 }

```

```

14 void ChangeString()

```

ChangeString()

修改字符串，对输入的字符串进行替换

```

15 {
16     int i;
17     for (i = 0; i < strlen(str); i++)
18     {
19         if ( _____2_____ )
20             str[i] = change[str[i] - 'A'] - 'a' + 'A';
21         else
22             _____3_____
23     }
24 }

```

```

25 int main()

```

```

26 {
27     int i;
28     cin >> str ;
29     cin >> change;
30     CheckChangeRule();
31     _____4_____
32     cout << str << endl;
33     return 0;
34 }

```

快快编程  
kkcoding.net

# 完善程序

关键语句

统一将字符的替换规则用  
小写字母表示

将大写转为小写

```
1  #include <iostream>
2  #include <cstring>
3  char change[26], str[5000];
4  using namespace std;
5  void CheckChangeRule()
6  {
7      int i;
8      for (i = 0; i < 26; i++)
9      {
10         if ( _____1_____ )
11             change[i] -= 'A' - 'a';
12     }
13 }
14 void ChangeString()
15 {
16     int i;
17     for (i = 0; i < strlen(str); i++)
18     {
19         if ( _____2_____ )
20             str[i] = change[str[i] - 'A'] - 'a' + 'A';
21         else
22             _____3_____
23     }
24 }
25 int main()
26 {
27     int i;
28     cin >> str ;
29     cin >> change;
30     CheckChangeRule();
31     _____4_____
32     cout << str << endl;
33     return 0;
34 }
```

快快编程  
kkcoding.net

# 完善程序

## 关键语句

```
1  #include <iostream>
2  #include <cstring>
3  char change[26], str[5000];
4  using namespace std;
5  void CheckChangeRule()
6  {
7      int i;
8      for (i = 0; i < 26; i++)
9      {
10         if ( _____1_____ )
11             change[i] -= 'A' - 'a';
12     }
13 }
14 void ChangeString()
15 {
16     int i;
17     for (i = 0; i < strlen(str); i++)
18     {
19         if ( _____2_____ )
20             str[i] = change[str[i] - 'A' - 'a' + 'A'];
21         else
22             _____3_____
23     }
24 }
25 int main()
26 {
27     int i;
28     cin >> str ;
29     cin >> change;
30     CheckChangeRule();
31     _____4_____
32     cout << str << endl;
33     return 0;
34 }
```

str[i]

当前要替换的第i个字符

change[str[i]]

当前要替换的第i个字符，  
按照替换规则替换为哪个  
字符

将change[x]转为大写

change[str[i]-'A']表示str[i]是大写字母替换后的字符。将小写转大写是因为替换规则统一小写，而str[i]替换后要与原文大小写一致

# 作业：完善程序

231 字符串的逆序

1987 数字删除

697 字符串替换