

LESSON_04 参考试题

一、 选择题

1. 以下哪个函数是用来在 string 类中删除字符 ()。

- A. empty()
- B. replace()
- C. substr()
- D. erase()

【答案】D

2. 执行以下代码，输出结果是()

- A. llo,C+
- B. llo,C
- C. lo,C+
- D. lo,C++

【答案】C

二、 判断题

1. (2023年6月)字符常量'\0'常用来表示字符串结束，它和字符常量'0'是不同的。

【答案】正确

2. (2023年9月)字符常量 '3' 的值和 int 类型常量 3 的值是相同的，只是占用的字节数不同。

【答案】错误

三、 编程题

1. 文本替换

【问题描述】

现在有一行英文单词组成的文本（区分大小写，以 "." 结尾），请你设计一个程序，输入要替换的单词和替换后的目标单词，将整篇文本所有需要替换的单词全部更改成目标单词，最后将全部文本输出。

【输入描述】

共三行。

第一行为一串单词文本 s ， s 的长度为 L ($0 < L < 200$)，单词之间用空格隔开；

第二行为要替换的单词 s_1 ， s_1 的长度为 L_1 ($0 < L_1 < 10$)；

第三行为目标单词 s_2 ， s_2 的长度为 L_2 ($0 < L_2 < 10$)。

【输出描述】

替换后新的一串单词文本。

【样例输入 1】

To ba or not to ba, that is a question.

ba
be

【样例输出 1】

To be or not to be, that is a question.

【参考代码】

```
#include <iostream>
using namespace std;
using namespace std;
int main()
{
    string s, s1, s2;
    getline(cin, s);
    getline(cin, s1);
    getline(cin, s2);
    for (int i = 0; (i = s.find(s1, i)) >= 0; i += s2.size()) {
        s.replace(i, s1.size(), s2);
    }
    cout << s;
    return 0;
}
```

2. （2023年6月）密码合规检测

【问题描述】

网站注册需要有用户名和密码，编写程序以检查用户输入密码的有效性。合规的密码应满足以下要求：

- 1、只能由 a-z 之间 26 个小写字母、A-Z 之间 26 个大写字母、0-9 之间 10 个数字以及!@#\$四个特殊字符构成。
- 2、密码最短长度：6 个字符，密码最大长度：12 个字符。
- 3、大写字母、小写字母和数字必须至少有其中两种，以及至少有四个特殊字符中的一个。

【输入描述】

输入一行不含空格的字符串。约定长度不超过 100。该字符串被英文逗号分隔为多段，作为多组被检测密码

【输出描述】

输出若干行，每行输出一组合规的密码。

输出顺序以输入先后为序，即先输入则先输出。

【样例输入】

seHJ12!@,sjdkffH\$123,sdf!@&12HDHa!,123&^YUhg@!

【样例输出】

seHJ12!@
sjdkffH\$123

【样例解释】

输入被英文逗号分为了四组被检测密码：“seHJ12!@”、“sjdkffH\$123”、“sdf!@&12HDHa!”、“123&^YUhg@!”。其中，“sdf!@&12HDHa!”长度超过 12 个字符，不合规；“123&^YUhg@!”包含四个特殊字符之外的字符“^”，不合规。

【参考代码】

```
#include <iostream>
using namespace std;
bool check(string str){
    if(str.length()>12||str.length()<6)return false;
    bool hasC=false,hasL=false,hasD=false,hasS=false;
    for(int i=0;i<str.length();i++){
        if('A'<=str[i]&&str[i]<='Z')hasC=true;
        else if('a'<=str[i]&&str[i]<='z')hasL=true;
        else if('0'<=str[i]&&str[i]<='9')hasD=true;
        else if(str[i]=='!'||str[i]=='@'||str[i]=='#'||str[i]=='$')
            hasS=true;
        else return false;
    }
    if(hasC+hasL+hasD<2)return false;
    if(!hasS)return false;
    return true;
}
int main(){
    string pwd;
    cin>>pwd;
    int pos=pwd.find(',');
    int begin=0;
    string line="";
    while(pos!=-1){
        for(int i=begin;i<pos;i++)line+=pwd[i];
        if(check(line))cout<<line<<endl;
        begin=pos+1;
        pos=pwd.find(',',begin);
        line="";
    }
    for(int i=begin;i<pwd.length();i++)line+=pwd[i];
    if(check(line))cout<<line;
    return 0;
}
```