

【华为OD机考 统一考试机试C卷】 密码输入检测（C++ Java JavaScript Python C语言）

华为OD机考:统一考试 C卷 + D卷 + B卷 +A卷

2023年11月份，华为官方已经将 华为OD机考：OD统一考试（A卷 / B卷）切换到 OD统一考试（C卷）和 OD统一考试（D卷）。

真题目录： [华为OD机考机试 真题目录（C卷 + D卷 + B卷 + A卷） + 考点说明](#)

专栏： [2023华为OD机试\(B卷+C卷+D卷\)（C++JavaJSPy）](#)

华为OD面试真题精选： [华为OD面试真题精选](#)

在线OJ： [点击立即刷题](#)，模拟真实机考环境

题目描述：密码输入检测（本题分值100）

给定用户密码输入流input，输入流中字符'<'表示退格，可以清除前一个输入的字符，请你编写程序，输出最终得到的密码字符，并判断密码是否满足如下的密码安全要求。

密码安全要求如下：

- 1.密码长度 ≥ 8 ;
- 2.密码至少需要包含1个大写字母;
- 3.密码至少需要包含1个小写字母;
- 4.密码至少需要包含1个数字;
- 5.密码至少需要包含1个字母和数字以外的非空白特殊字符

注意空串退格后仍然为空串，且用户输入的字符串不包含'<'字符和空白字符。

输入描述

用一行字符串表示输入的用户数据，输入的字符串中'<'字符标识退格，用户输入的字符串不包含空白字符，例如： `ABc<c89%000<`

输出描述

输出经过程序处理后，输出的实际密码字符串，并输出改密码字符串是否满足密码安全要求。两者间由','分隔， 例如： `ABc89%00,true`

示例1

输入

```
1 | ABc<c89%000<
```

输出

```
1 | ABc89%00,true
```

说明

解释：多余的C和0由于退格被去除,最终用户输入的密码为ABc89%00，且满足密码安全要求，输出true

解题思路

1. 始化了五个变量： `result` （一个空字符串用于存储处理后的输入）， `is_big`， `is_small`， `is_num` 和 `is_spec` （四个布尔变量，用于检查处理后的字符串是否包含大写字母、小写字母、数字和特殊字符）。
2. 接下来，代码遍历输入的字符串中的每一个字符。对于每一个字符，代码首先检查它是否是'<'。如果是，那么代码将删除 `result` 字符串的最后一个字符（如果存在的话）。否则，代码将执行以下操作：
 - 将字符添加到 `result` 字符串中。
 - 如果字符是数字，并且 `is_num` 当前为False，那么将 `is_num` 设置为True。
 - 如果字符是小写字母，并且 `is_small` 当前为False，那么将 `is_small` 设置为True。
 - 如果字符是大写字母，并且 `is_big` 当前为False，那么将 `is_big` 设置为True。
 - 如果字符是特殊字符（即，它不是数字、小写字母、大写字母或空格），并且 `is_spec` 当前为False，那么将 `is_spec` 设置为True。

3. 在遍历完输入的字符串后，代码将检查 `result` 字符串是否满足以下条件：长度大于等于8，并且包含大写字母、小写字母、数字和特殊字符。如果满足这些条件，那么 `flag_res` 将被设置为True，否则，它将被设置为False。
4. 最后，代码将输出 `result` 字符串和 `flag_res` 的值。这两个值之间用逗号分隔，`flag_res` 的值被转换为字符串格式。

总的来说，这段代码的主要目标是对用户的输入进行处理，并检查处理后的结果是否满足特定的条件。

C++

```
1  #include <iostream>
2  #include <string>
3  #include <cctype>
4
5  int main() {
6      // 创建一个字符串来读取用户的输入
7      std::string input;
8      std::getline(std::cin, input);
9
10     // 创建一个字符串来构建结果字符串
11     std::string result = "";
12
13     // 创建四个布尔变量来检查输入字符串中是否包含大写字母、小写字母、数字和特殊字符
14     bool isBig = false;
15     bool isSmall = false;
16     bool isNum = false;
17     bool isSpec = false;
18
19     // 遍历输入字符串中的每一个字符
20     for (char c : input) {
21         // 如果字符是'<'，则删除结果字符串的最后一个字符
22         if (c == '<') {
23             if (!result.empty()) {
24                 result.pop_back();
25             }
26         } else {
27             // 否则，将字符添加到结果字符串中
28             result.push_back(c);
29         }
30     }
31 }
```

```

51
52
53 // 遍历输入字符串中的每一个字符
54 for (char c : result) {
55     // 检查字符是否是数字
56     if (isdigit(c)) {
57         isNum = true;
58     }
59     // 检查字符是否是小写字母
60     else if (islower(c)) {
61         isSmall = true;
62     }
63     // 检查字符是否是大写字母
64     else if (isupper(c)) {
65         isBig = true;
66     }
67     // 检查字符是否是特殊字符
68     else {
69         isSpec = true;
70     }
71 }
72
73 // 检查结果字符串是否满足长度大于等于8, 并且包含大写字母、小写字母、数字和特殊字符
74 bool flagRes = result.size() >= 8 && isNum && isSmall && isBig && isSpec;
75
76 // 输出结果字符串和检查结果
77 std::cout << result << ", " << std::boolalpha << flagRes << std::endl;
78
79 return 0;
80 }

```

Java

```

1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) {
5         // 创建一个Scanner对象来读取用户的输入
6         Scanner scanner = new Scanner(System.in);
7         // 读取一行输入
8         String input = scanner.nextLine();
9     }

```

```

9      // 创建一个StringBuilder对象来构建结果字符串
10     StringBuilder result = new StringBuilder();
11     // 创建四个布尔变量来检查输入字符串中是否包含大写字母、小写字母、数字和特殊字符
12     boolean isBig = false;
13     boolean isSmall = false;
14     boolean isNum = false;
15     boolean isSpec = false;
16
17     // 遍历输入字符串中的每一个字符
18     for (char c : input.toCharArray()) {
19         // 如果字符是'<', 则删除结果字符串的最后一个字符
20         if (c == '<') {
21             if (result.length() > 0) {
22                 result.deleteCharAt(result.length() - 1);
23             }
24         } else {
25             // 否则, 将字符添加到结果字符串中
26             result.append(c);
27
28         }
29     }
30
31     // 遍历输入字符串中的每一个字符
32     for (int i = 0; i < result.length(); i++) {
33         char c = result.charAt(i);
34
35         // 检查字符是否是数字
36         if (Character.isDigit(c)) {
37             isNum = true;
38         }
39         // 检查字符是否是小写字母
40         else if (Character.isLowerCase(c)) {
41             isSmall = true;
42         }
43         // 检查字符是否是大写字母
44         else if (Character.isUpperCase(c)) {
45             isBig = true;
46         }
47         // 检查字符是否是特殊字符
48         else {
49

```

```

50         isSpec = true;
51     }
52 }
53
54 // 检查结果字符串是否满足长度大于等于8, 并且包含大写字母、小写字母、数字和特殊字符
55 boolean flagRes = result.length() >= 8 && isNum && isSmall && isBig && isSpec;
56
57 // 输出结果字符串和检查结果
58 System.out.println(result + "," + flagRes);
59 }
60 }

```

javaScript

```

1  const readline = require('readline').createInterface({
2      input: process.stdin,
3      output: process.stdout
4  });
5
6  readline.on('line', input => {
7      let result = '';
8      let isBig = false;
9      let isSmall = false;
10     let isNum = false;
11     let isSpec = false;
12
13     for (let c of input) {
14         if (c === '<') {
15             result = result.slice(0, -1);
16         } else {
17             result += c;
18         }
19     }
20
21     for (let c of result) {
22         if (/[0-9]/.test(c)) {
23             isNum = true;
24         } else if (/[a-z]/.test(c)) {
25             isSmall = true;
26         }
27     }
28
29     console.log(result + "," + (isNum && isSmall && isBig && isSpec));
30 }

```

```

46     } else if (/[A-Z]/.test(c)) {
47         isBig = true;
48     } else {
49         isSpec = true;
50     }
51 }
52
53 let flagRes = result.length >= 8 && isNum && isSmall && isBig && isSpec;
54
55 console.log(result + "," + flagRes);
56
57 readline.close();
58 });

```

Python

```

1  # 读取用户的输入
2  input_str = input()
3
4  # 创建一个字符串来构建结果字符串
5  result = ""
6
7  # 创建四个布尔变量来检查输入字符串中是否包含大写字母、小写字母、数字和特殊字符
8  is_big = False
9  is_small = False
10 is_num = False
11 is_spec = False
12
13 # 遍历输入字符串中的每一个字符
14 for c in input_str:
15     # 如果字符是 '<', 则删除结果字符串的最后一个字符
16     if c == '<':
17         result = result[:-1]
18     else:
19         # 否则, 将字符添加到结果字符串中
20         result += c
21
22 # 遍历输入字符串中的每一个字符
23 for c in result:
24     # 检查字符是否是数字
25

```

```

25     if c.isdigit():
26         is_num = True
27     # 检查字符是否是小写字母
28     elif c.islower():
29         is_small = True
30     # 检查字符是否是大写字母
31     elif c.isupper():
32         is_big = True
33     # 检查字符是否是特殊字符
34     else:
35         is_spec = True
36
37 # 检查结果字符串是否满足长度大于等于8, 并且包含大写字母、小写字母、数字和特殊字符
38 flag_res = len(result) >= 8 and is_num and is_small and is_big and is_spec
39
40 # 输出结果字符串和检查结果
41 print(result + "," + str(flag_res).lower())

```

C语言

```

1  #include <stdio.h>
2  #include <string.h>
3  #include <ctype.h> // 包含字符处理函数
4
5  #define MAX_LEN 1000 // 定义最大输入长度
6
7  int main() {
8      char input[MAX_LEN];
9      gets(input);
10
11
12     // 定义用于构建最终密码的字符串数组
13     char result[MAX_LEN];
14     int length = 0; // 结果字符串的当前长度
15
16     // 定义四个布尔变量, 分别用来标记是否含有大写字母、小写字母、数字和特殊字符
17     int hasUpper = 0;
18     int hasLower = 0;
19     int hasDigit = 0;
20     int hasSpecial = 0;
21

```



```

21
22 // 遍历输入字符串
23 for (int i = 0; input[i] != '\0'; ++i) {
24     if (input[i] == '<') {
25         // 如果是退格符号, 且结果字符串长度大于0, 则删除最后一个字符
26         if (length > 0) {
27             length--;
28         }
29     } else {
30         // 否则添加字符到结果字符串
31         result[length++] = input[i];
32     }
33 }
34 result[length] = '\0'; // 字符串结束标志
35
36 // 再次遍历结果字符串, 检查密码安全要求
37 for (int i = 0; i < length; ++i) {
38     if (isupper(result[i])) hasUpper = 1; // 检查大写字母
39     else if (islower(result[i])) hasLower = 1; // 检查小写字母
40     else if (isdigit(result[i])) hasDigit = 1; // 检查数字
41     else hasSpecial = 1; // 检查特殊字符
42 }
43
44 // 判断是否满足密码安全要求
45 int isValid = length >= 8 && hasUpper && hasLower && hasDigit && hasSpecial;
46
47 // 输出处理后的密码及其是否符合安全要求
48 printf("%s,%s\n", result, isValid ? "true" : "false");
49
50 return 0;
51 }

```

完整用例

用例1

Aa1<2#<3\$

用例2

abcde<FGH1!

用例3

aB1#<2\$<3%

用例4

Aa1#<2\$<3%<

用例5

aaaaaaA1!

用例6

AAAAAAa1!

用例7

AaBbCcDd!

用例8

Aa1Bb2Cc3

用例9

Aa1#Bb2\$Cc3%

用例10

Aa1#<Bb2\$<Cc3%<

文章目录

[华为OD机考:统一考试 C卷 + D卷 + B卷 +A卷](#)

[题目描述：密码输入检测（本题分值100）](#)

[输入描述](#)

[输出描述](#)

示例1

解题思路

C++

Java

javaScript

Python

C语言

完整用例

用例1

用例2

用例3

用例4

用例5

用例6

用例7

用例8

用例9

用例10

机考C卷真题

华为OD

