

# 【华为OD机考 统一考试机试C卷】字符串筛选排序（C++ Java JavaScript Python C语言）

## 华为OD机考:统一考试 C卷 + D卷 + B卷 +A卷

目前在考C卷，经过两个月的收集整理，**C卷真题已基本整理完毕**

抽到原题的概率为2/3到3/3，**也就是最少抽到两道原题。请注意：大家刷完C卷真题，最好要把B卷的真题刷一下，因为C卷的部分真题来自B卷。**

另外订阅专栏还可以联系笔者开通在线 **OJ** 进行刷题，提高刷题效率。

**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

输入一个由N个大小写字母组成的字符串

按照ASCII码值从小到大进行排序

查找字符串中第K个最小ASCII码值的字母(k>=1)

输出该字母所在字符串中的位置索引(字符串的第一个位置索引为0)

k如果大于字符串长度则输出最大ASCII码值的字母所在字符串的位置索引

如果有重复字母则输出字母的最小位置索引

## 输入描述

第一行输入一个由大小写字母组成的字符串

第二行输入k，k必须大于0，k可以大于输入字符串的长度

## 输出描述

输出字符串中第k个最小ASCII码值的字母所在字符串的位置索引

k如果大于字符串长度则输出最大ASCII码值的字母所在字符串的位置索引

如果第k个最小ASCII码值的字母存在重复 则输出该字母的最小位置索引

用例

输入	AbCdeFG 3
输出	5
说明	<ul style="list-style-type: none"><li>根据ASCII码值排序，第三个ASCII码值的字母为FF在字符串中位置索引为5(0为字符串的第一个字母位置索引)</li></ul>

C++

```
1 |
2 | #include <iostream>
3 | #include <algorithm>
4 | #include <cstring>
5 |
6 | using namespace std;
7 |
8 | int main() {
9 |     string str;
10 |    cin >> str;
11 |    int k;
12 |    cin >> k;
13 |    char chars[str.length()];
14 |    strcpy(chars, str.c_str());
15 |    sort(chars, chars+str.length());
16 |
17 |    if (k > str.length()) {
18 |        k = str.length();
19 |    }
20 |
21 |    char tar = chars[k - 1];
22 |
23 |    cout << str.find(tar) << endl;
24 |    return 0;
25 | }
```

## java

```
1 import java.util.Arrays;
2 import java.util.Scanner;
3
4 public class Main {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         String str = scanner.nextLine();
8         int k = scanner.nextInt();
9         char[] chars = str.toCharArray();
10        Arrays.sort(chars);
11
12        if (k > chars.length) {
13            k = chars.length;
14        }
15
16        char tar = chars[k - 1];
17
18        System.out.println(str.indexOf(tar));
19    }
20 }
```

## javaScript

```
1 const readline = require('readline');
2
3 const rl = readline.createInterface({
4     input: process.stdin,
5     output: process.stdout
6 });
7
8 rl.on('line', (str) => {
9     rl.on('line', (k) => {
10         k = parseInt(k);
11         let chars = str.split('').sort();
12         if (k > str.length) {
13             k = str.length;
14         }
15         let tar = chars[k - 1];
```

```
16     console.log(str.indexOf(tar));
17     rl.close();
18 });
19 });
```

## python

```
1 # 导入Python的内置模块
2 import sys
3
4 def main():
5     # 读取一行输入并去除两端的空白字符 (例如换行符)
6     str = input().strip()
7     # 读取下一行输入, 转换为整数
8     k = int(input().strip())
9     # 将字符串转换为字符列表
10    chars = list(str)
11    # 对字符列表进行排序
12    chars.sort()
13
14    # 如果k大于字符列表的长度, 则将k设置为字符列表的长度
15    if k > len(chars):
16        k = len(chars)
17
18    # 获取排序后的第k个字符 (由于索引从0开始, 所以是第k-1个位置的字符)
19    tar = chars[k - 1]
20
21    # 输出目标字符在原始字符串中的索引位置
22    # 注意: index方法返回的是第一个找到的索引位置
23    print(str.index(tar))
24
25 # 调用main函数
26 if __name__ == "__main__":
27     main()
```

## C语言

```
1 #include <stdio.h>
2 #include <string.h>
3
```

```
4  #include <stdlib.h>
5
6  #define MAX_STR_LEN 1001
7
8  int compare(const void *a, const void *b) {
9      return (*(const char *)a - *(const char *)b);
10 }
11
12 int findFirstOccurrence(const char *str, char ch) {
13     // 查找字符ch在字符串str中的第一个出现位置
14     for (int i = 0; str[i] != '\0'; i++) {
15         if (str[i] == ch) {
16             return i; // 返回找到的位置索引
17         }
18     }
19     return -1; // 如果字符不在字符串中, 返回-1
20 }
21
22 int main() {
23     char str[MAX_STR_LEN];
24     int k;
25     scanf("%s", str); // 输入一个由大小写字母组成的字符串
26     scanf("%d", &k); // 输入k
27
28     int len = strlen(str);
29     char sorted_str[MAX_STR_LEN];
30     strcpy(sorted_str, str);
31     qsort(sorted_str, len, sizeof(char), compare); // 根据ASCII码值排序
32
33     if (k > len) {
34         k = len;
35     }
36
37     char target = sorted_str[k - 1];
38     int index = findFirstOccurrence(str, target);
39
40     printf("%d\n", index); // 输出所求字符的位置索引
41     return 0;
}
```

完整用例

用例1

AbCdeFG  
3

用例2

zZyYxX  
6

用例3

HelloWorld  
10

用例4

HelloWorld  
15

用例5

aBcDeFgHiJkLmNoPqRsTuVwXyZ  
5

用例6

fAdDAkBbBq  
1

用例7

PythonIsFun  
4

用例8

QuickBrownFox  
8

用例9

LazyDog  
5

用例10

aBcDeFgHiJkLmNoPqRsTuVwXyZ  
12

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
  - 输入描述
  - 输出描述
  - 用例
    - 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

# 机考真题 华为OD



CSDN @算法大师