

高级程序设计与实验随堂测试四

- 全局变量被视为不良的编程风格。每声明一个全局变量，相应题目将扣除 10% 的分数。静态 (static) 变量视为全局变量处理。
- 本次测试重点考查向量和字符串的使用，不允许使用 C 风格的数组。每道题目中，每声明一个数组，该题目扣除 30% 的分数。

1. (运行时间, 10 分) 分析以下两个函数的运行时间, f 的运行时间用大 $\Theta(\)$ 记号表示为整数 a 的函数, g 的运行时间用大 $\Theta(\)$ 记号表示为整数 n 的函数. 本题提交一个.txt 文件, 文件的第一行是 f 的运行时间, 第二行是 g 的运行时间, 只需写 $\Theta(\)$ 括号内的内容。

```
int f(unsigned int a)
{
    int b = 1, count = 0;
    while(b < a)
    {
        b *= 2;
        count++;
    }
    return count;
}

int g(const vector<int>& nums) {
    int maxLength = 0;
    int n = nums.size();

    int left = 0;
    while (left < n) {
        if (nums[left] != 0) {
            ++left;
            continue;
        }

        int right = left;
        while (right < n && nums[right] == 0) {
            ++right;
        }
        maxLength = max(maxLength, right - left);

        left = right;
    }
    return maxLength;
}
```

2. (数元音个数, 10分) 写一个 C++ 函数, 它的输入是一个字符串, 输出是该字符串中元音字母的个数。元音是指 a, e, i, o, u 这五个字母。注意, 大写和小写字母都要计入。

函数格式: `int nVowel (const string & s);`

3. (解码, 10分) 给定一个 $N \times N$ 的 0 - 1 字符阵, 我们按以下规则生成一个压缩码, 压缩码是一系列正整数 A_1, A_2, \dots ,

从字符阵的第一行第一个字符开始计算, 按书写顺序 (即从左到右、由上至下), A_1 表示连续有几个 0, A_2 表示接下来连续有几个 1, A_3 表示再接下来连续有几个 0, A_4 接着连续几个 1, 以此类推.....

例如: 以下汉字点阵图案:

0001000

0001000

0001111

0001000

0001000

0001000

1111111

对应的压缩码是: 3 1 6 1 6 4 3 1 6 1 6 1 3 7

写一个 C++ 函数, 输入 N 和压缩码 A , 输出它的字符阵。其中, 压缩码以 `vector<int>` 的形式输入, 输出则是一个有 N 个元素的 `vector<string>`, 其中每个 `string` 的长度为 N 。你可以假定 A 中的元素之和恰好等于 $N \times N$ 。

函数格式: `vector<string> decode(int N, const vector<int> & A);`

下面这段主函数可以帮你测试：

```
#include <string>

#include <vector>

#include <iostream>

using namespace std;

int main()

{

    vector<string> B = decode(7, {3, 1, 6, 1, 6, 4, 3, 1, 6, 1, 6, 1, 3, 7});

    for (int i = 0; i < B.size(); i++)

        cout << B[i] << endl;

    return;

}
```

4. (Bonus question, 7 分) 写一个 C++ 函数，输入一个含有 n 个正整数的数列和一个正整数 T 。找出该数列中满足其和至少是 T 的长度最小的连续子数列，并返回其该子数列的长度。如果不存在符合条件的子数组，返回 0。

示例：输入 $T = 7$, $nums = [2,3,1,2,4,3,6,5]$ ，输出 2（子数组 $[4,3]$ 是该条件下的长度最小的子数组）。

函数的运行时间必须是 $O(n)$ 的，否则本题不得分。

函数格式： `int minSequence(int T, const vector<int> & nums);`