

数据结构 A

作业 1 参考答案

作业情况

教材：数据结构教程（C++ 语言描述）李春葆等

题目范围：绪论、算法分析

邮箱：wjyyy1@126.com

授课教师：彭蓉 教授

助教：王骏骁

习题 1 单选题

某算法的空间复杂度为 $O(1)$ ，则（ ）。

- A. 该算法执行所需辅助空间大小与问题规模 n 无关
- B. 该算法执行所需全部空间大小与问题规模 n 无关
- C. 该算法执行不需要任何辅助空间
- D. 该算法执行不需要任何空间

答案：A

解析：参考课件「第 1 讲-绪论」第 87 页，一个算法的存储量包括形参所占空间和临时变量所占空间。在对算法进行存储空间分析时，只考察临时变量所占空间。

参考课本 25 页，空间复杂度是对一个算法在执行过程中临时占用的存储空间

的量度。因此考虑带辅助空间的选项。其中 $O(1)$ 表示线性复杂度，例如使用了常数个辅助变量，也就是辅助空间大小与问题规模 n 无关，故选 A。

习题 2 求时间复杂度

已知 n 为正整数，以下算法的时间复杂度为 ()。

```
1 void fun(int n)
2 {
3     int i,j,s=0;
4     for(i=1;i<=n;i=i*3)
5         for(j=i/3;j<=i;j++)
6             s=s+j;
7 }
```

答案: $O(n)$

解析: 分析算法复杂度, 就是考虑这段代码一共执行了多少次语句。

方法 1: 本题为填空题, 可以直接本地代码跑出结果。

```
1 void fun(int n)
2 {
3     int i,j,s=0;
4     int cnt=0;
5     for(i=1;i<=n;i=i*3)
6         for(j=i/3;j<=i;j++)
7             {
8                 s=s+j;
9                 cnt++;
10            }
11     cout<<n<<" "<<cnt<<endl;
12 }
```

输入 $n = 100$, 得到结果 $cnt = 86$; 输入 $n = 1000$, 得到结果 $cnt = 736$ 。

大致分析出算法复杂度和 n 线性相关, 复杂度不会超过 $O(n)$ 。熟练后可以推测出结果为 $O(n)$ 。

方法 2: 求和算法分析。外层循环执行了 $\log_3 n$ 次, 每次循环的时候的 i 带入内层循环。现在在进行第几次外层循环, i 就是 3 的几次方。也就是说, 第 k

次外层循环时, $i = 3^k$ 。

因此内层的 j 从 $i/3$ 到 i 一共是 $\frac{2}{3}i + 1$ 次, 复杂度中常数忽略之后可认为执行了 $O(i)$ 次。

所以算法分析的结果是:

$$\begin{aligned} T(n) &= \sum_{k=1}^{\log_3 n} \left(\frac{2}{3}i + 1 \right) \\ &\approx \sum_{k=1}^{\log_3 n} i \\ &= \sum_{k=1}^{\log_3 n} 3^k \\ &= \frac{1 - 3^{\log_3 n}}{1 - 3} \text{ (等比数列求和)} \\ &= O(n) \end{aligned}$$

习题 3 最大因子**【问题描述】**

给定一组共 n ($1 \leq n \leq 5000$) 个整数，整数的取值范围为 1 到 20000，请确定具有最大素数因子的整数（请记住，素数只有因子 1 和自身，整数 7 是素数，而整数 6 可以被 2 和 3 整除，它不是素数）。

【输入形式】

第一行为单个整数 n ，接下来共 n 个整数，每行包含一个整数。

【输出形式】

在一行中输出具有最大素数因子的那个整数，如果有多个，则输出最早出现在输入文件中的一个。

【样例输入】

4
36
38
40
42

【样例输出】

38

【样例说明】

测试数据的文件名为 in.txt

【评分标准】

该题目有 5 个测试用例，每通过一个测试用例，得 20 分。

答案：

```
1 #include<iostream>
2 using namespace std;
3 // 定义函数prime，返回x的最大因子
4 int prime(int x)
5 {
6     for(int i=x;i>=2;i--)
7     {
```

```
8      //如果i是x的因子, 就判断i是不是质数
9      if(x%i==0)
10     {
11         bool is_prime=true;
12         for(int j=2;j*j<=i;j++)
13             if(i%j==0)
14                 is_prime=false;
15         if(is_prime)
16             return i;
17     }
18 }
19 return 0;
20 }
21 int main()
22 {
23     int n,ans=0,max_prime=0;
24     //输入n, 定义ans和max_prime存储当前答案
25     //和答案所拥有的最大因子
26     ifstream fin("in.txt");
27     //使用文件流从in.txt读取数据
28     fin>>n;
29     for(int i=1;i<=n;i++)
30     {
31         //读入n次数字x
32         int x;
33         fin>>x;
34         //从函数中获取x的最大因子
35         int new_prime=prime(x);
36         //如果x的最大因子比之前出现过数的最大因子更大
37         //就更新答案和最大因子
38         if(new_prime>max_prime)
```

```
39         {
40             ans=x;
41             max_prime=new_prime;
42         }
43     }
44     cout<<ans;
45     return 0;
46 }
```

解析：对于 n 个数据，需要找到其中所有数中拥有最大质数因子的一个。所以算法流程为：

1. 对每个数找出它的因子，并返回其中最大的质数。（函数 `int prime(x)`）
2. 比较所有返回值，拥有最大返回值，并最早出现的一个就是答案。

注意算法复杂度，如果发生时间超限，可以考虑优化算法。例如提前把所有质数存储在一个列表里。

习题 4 回文数

【问题描述】

判断一个整数是否是回文数。例如，121 是回文数，而 -121 不是回文数。题目要求设计如下满足要求的函数：

```
1 class Solution {  
2 public:  
3     bool isPalindrome(int x)  
4     { ... }  
5 };
```

【输入形式】

每个测试用例输入一行，是一个整数。

【输出形式】

对于每个测试用例输出一行，输出 0 表示不是回文数，或者输出 1 表示是回文数。

【样例输入】

12345

【样例输出】

0

【样例说明】

测试用例中的每一行代表一个待测试的整数，测试数据的文件名为 in.txt。

【评分标准】

该题目有 10 个测试用例，每通过一个测试用例，得 10 分。

答案：

```
1 class Solution {  
2     public:  
3         bool isPalindrome(int x)  
4         {  
5             //如果是负数，负号不会有对应，直接返回否
```

```
6         if(x<0)
7             return false;
8         // 定义可变数组/链表，每次放x的一位进去
9         vector<int> v;
10        // 取出x的最低位，并除以10，就能取出所有位
11        while(x>0)
12        {
13            v.push_back(x%10);
14            x/=10;
15        }
16        // 从前往后数第i位，应该对应从后往前第i位
17        // 下标从0开始，到v.size()-1结束
18        for(int i=0;i<v.size();i++)
19            if(v[i]!=v[v.size()-i-1])
20                return false;
21        return true;
22    }
23};
```


总结

题目：绪论

日期：2024 年 3 月 11 日

批改人：王骏骁

邮箱：wjyyy1@126.com

各位同学写填空题也需要注意过程，作业中的填空题如果答案错误也会酌情给分。

习题 1：选择题注意找题目中的关键词，并注意参考上课讲的内容。

习题 2：大部分同学没有分析对复杂度，建议使用数学推导，并使用代码验证。另外，课本中的 O 均为大写，代表复杂度上界，小写 o 有其他含义。本次作业中写小写 o 的同学被扣掉 1 分（该题满分 10 分）。

习题 3：注意本次提交需要用到文件输入，而输出是直接用 `cout` 或 `printf` 给到标准输出中的。对基础代码结构不太了解的同学需要注意复习程序设计，或在群里提问。

习题 4：注意本题需要处理 $n < 0$ 的情况。建议使用题目要求的格式进行代码补全。