

LESSON_15 参考试题

一、 选择题

1. (2023年9月) 对包含 n 个元素的数组进行**冒泡排序**，平均时间复杂度一般为()。
A. $O(n)$
B. $O(n \log n)$
C. $O(n^2)$
D. 以上都不正确

【答案】C

2. 在对 n 个元素进行**冒泡排序**的过程中，最好情况下的时间复杂度为()。
A. $O(1)$
B. $O(\log_2 n)$
C. $O(n^2)$
D. $O(n)$

【答案】D

3. 在对 n 个元素进行直接**插入排序**的过程中，算法的空间复杂度为()。
A. $O(1)$
B. $O(\log_2 n)$
C. $O(n^2)$
D. $O(n)$

【答案】A

二、 判断题

1. (2023年9月) 对 N 个元素的数组执行插入排序算法，通常的时间复杂度为 $O(N^2)$ 。

【答案】正确

2. 插入排序平均时间复杂度为 $O(n^2)$ ，空间复杂度为1。

【答案】正确

3. (2023年12月) 用递归法求的 n 阶乘，时间复杂度是 $O(n)$ 。

【答案】正确

三、 编程题

- 1.信息老师的前缀和

【问题描述】

运用前缀和思想进行数据预处理，能够避免数据的重复计算，且大大降低了算法的时间复杂度。这天，狐狸老师想考一考大家对前缀和的掌握情况，于是出了一道题：求 $a \sim b$ 之间 1 出现的次数。

【输入描述】

共 $n+1$ 行；

第一行，一个整数 n ($1 \leq n \leq 10^5$)，表示查询次数。

接下来 n 行，每行包含两个整数 a 和 b ($1 \leq a < b \leq 10^5$)，数与数之间以一个空格隔开。

【输出描述】

n 行，每行一个整数，分别表示每次循环时 1 出现的次数。

【输入样例 1】

```
3
1 10
20 30
100 110
```

【输出样例 1】

```
2
1
13
```

【参考代码】

```
#include <bits/stdc++.h>
using namespace std;
int sum[100005];
int func(int x){
    int cnt=0;
    while(x!=0){
        if(x%10==1)cnt++;
        x/=10;
    }
    return cnt;
}
int main(){
    for(int i=1;i<=100000;i++){
        int cnt=func(i);
        sum[i]=sum[i-1]+cnt;
    }
    int n;
    cin>>n;
    int a,b;
    for(int i=0;i<n;i++){
        cin>>a>>b;
        cout<<sum[b]-sum[a-1]<<endl;
    }
    return 0;
}
```