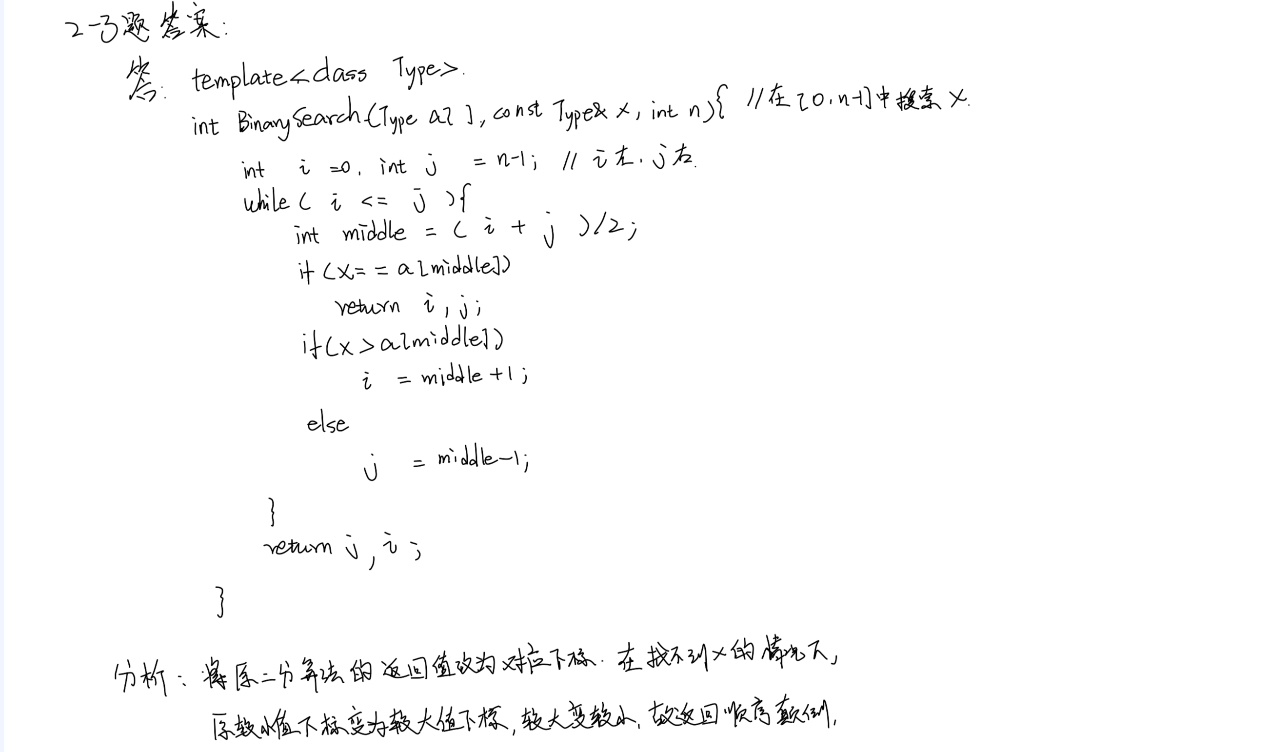
算法分析第二次作业

算法分析题

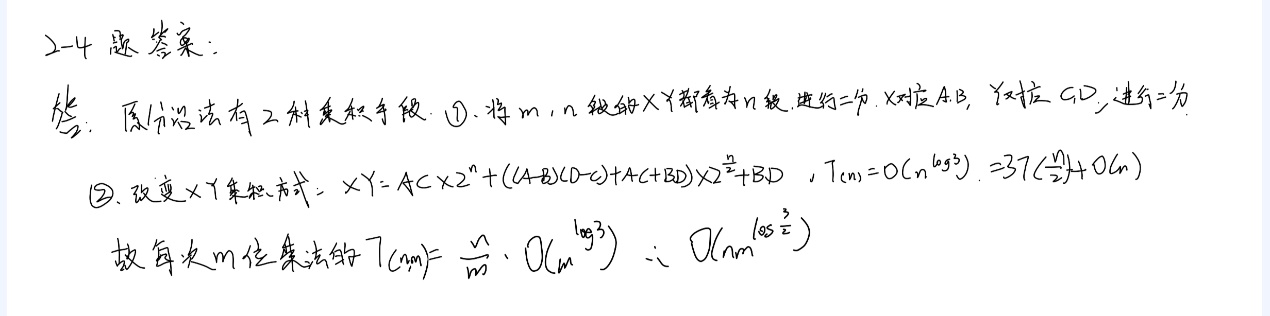
**2-3题答案：**

**答：**



**2-4题答案：**

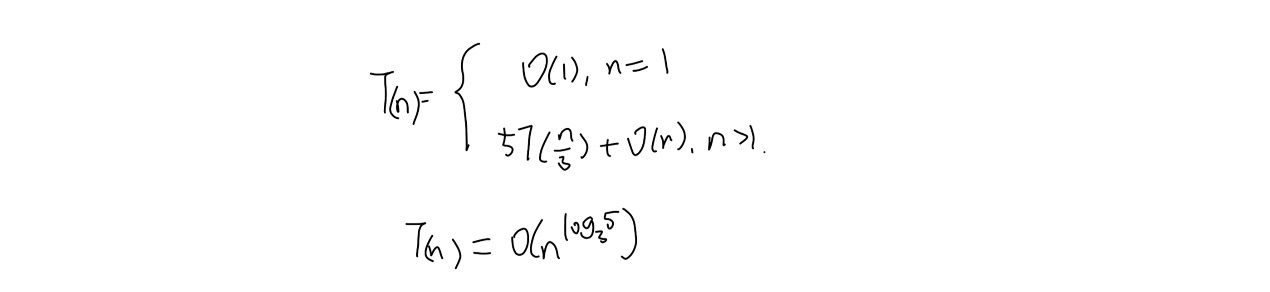
**答：**



**2-5题答案：**

**答：**

该题分治算法需要 5 次 n/3 位整数乘法。分割以及合并所需要的加减法和移位运算时间为 O(n)。设 T(n)是算法所需的计算时间，则：



u = a × 102(n/3) + b × 10 (n/3) + c

v = d × 102(n/3) + e × 10 (n/3) + f

P1 = a × d

P2 = c × f

P3 = (a+b) × (d+e)

P4 = (b+c) × (e+f)

P5 = (a+b+c) × (d+e+f) P6 = P1 + P2 + P3 + P4-P5

uv = P2 + (P4-P6-P2)×10 (n/3) + (P5-P3-P4+P6+P6)×102(n/3) + (P3-P6-P1)× 103(n/3) + P1×104(n/3)

**2-8题答案：**

**答：逆置子空间a[0:k-1]的时间复杂度为O(k),逆置子空间a[k:n1]的时间复杂度为O(n-k),而逆置整个数组的时间复杂度为O(n)，所以总的时间复杂度是O(k) + O(n-k) + O(n) = O(n)，逆置时用到一个临时变量，所以只用到O(1)的辅助空间。**

**2-9题答案：**

**答：**

**使用：采用双指针法，分别在两个已排序的子数组上移动指针，比较指针所指元素的大小，按顺序将较小的元素放入原数组的合适位置，直至其中一个子数组的元素全部处理完，再将另一个子数组剩余元素依次放入原数组.**

**该算法的平均时间复杂度为O(n)，没有用到辅助空间，只有循环换位时用到一个O(1)的辅助空间。**

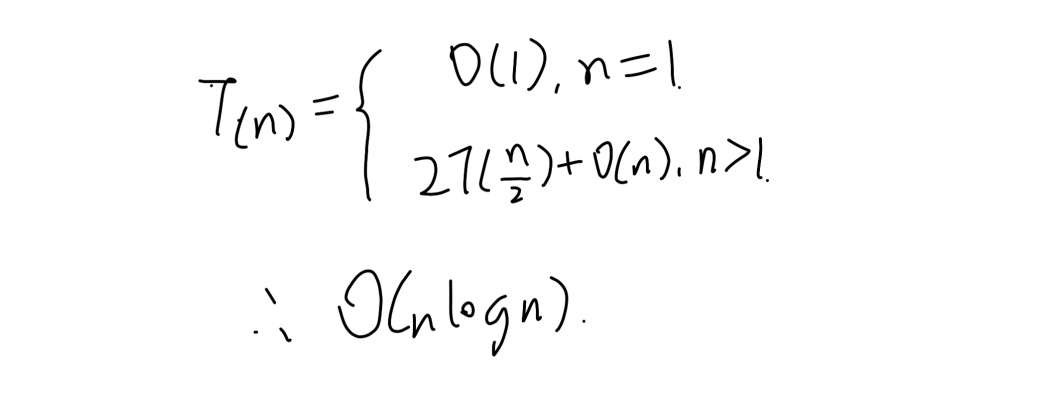
算法实现题

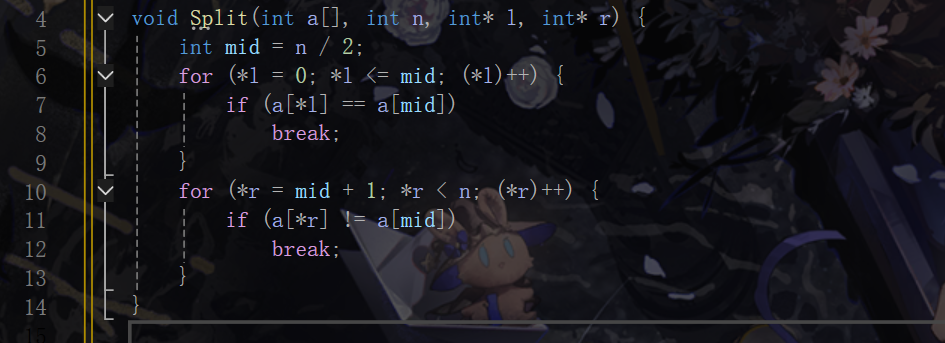
**2-1题答案：**

**答：**

**排序数组，对于一个有序的长度为n的数组S，可以通过中位数将其分为三部分：中位数，中位数以左部分，中位数以右部分。用两个指针left和right分别指向中位数第一次出现的位置和最后一次出现的位置，right-left+1即为中位数的重数count，并且根据要求更新。 中位数左边部分长度大于maxcount，递归。中位数右边部分长度大于maxcount，递归。**

**算法分析：排序数组的时间复杂度为O(nlogn),寻找中位数的时间复杂度为O(n)，而算法中分治递归过程的时间复杂度可以表示为：**

****

****

****

**2-7题答案：**

**答：**

**可以将问题分解成把n 个元素划分为m 个非空子集的集合，并求其个数总和，其中 m=1，2……，n。**

**递推公式：**

**Sum[n][m] = Sum[n-1][m-1] + m \* Sum[n-1][m]**

**边界情况：**

**Sum[0][j]=0、Sum[i][1]=1、Sum[i][i]=1。**

**双重for循环：时间复杂度为O (n2)。**

