**算法分析与设计**

**作业2**

王汝芸

201711010202

计工本1702

2019年9月10日

1. **分别写出、、的递归表达式**

01 '''

02 n!递归表达式

03 '''

04 **def** factorial**(**n**):**

05 **if** n**==**0**:**

06 **return** 1

07 **else:**

08 **return** n**\***factorial**(**n**-**1**)**

01 '''

02 sum(i)递归表达式

03 '''

04 **def** **sum(**n**):**

05 **if** n**==**0**:**

06 **return** 0

07 **else:**

08 **return** n**+sum(**n**-**1**)**

01 '''

02 sum(x\_i)递归表达式

03 '''

04 **def** **sum(**x**,**n**):**

05 **if** n**-**1**==**0**:**

06 **return** x**[**0**]**

07 **else:**

08 **return** x**[**n**-**1**]+sum(**x**,**n**-**1**)**

09

10 x **= [**1**,**2**,**3**,**4**,**5**,**6**,**7**,**8**,**9**,**10**]**

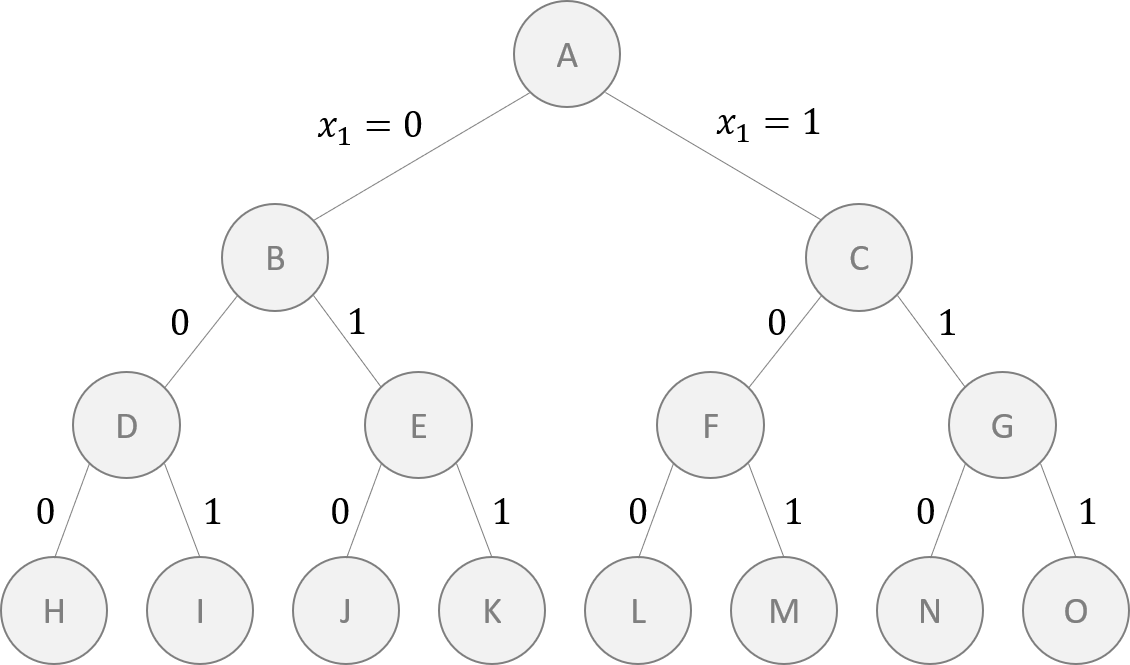
11 **print(sum(**x**,len(**x**)))**

1. **给出0-1背包问题的最优化问题和判定性问题的定义**

最优化问题：给定n个重量为，，…，价值为，，…，的物品和容量为W的背包，其中且物品不可分割，问装入哪些物品可以获得最大的价值？

判定性问题：给定n个重量为，，…，价值为，，…，的物品和容量为W的背包，其中且物品不可分割，和价值，问是否存在价值不低于的总重量不超过W的物品组合？

1. **画出n=3的子集树**

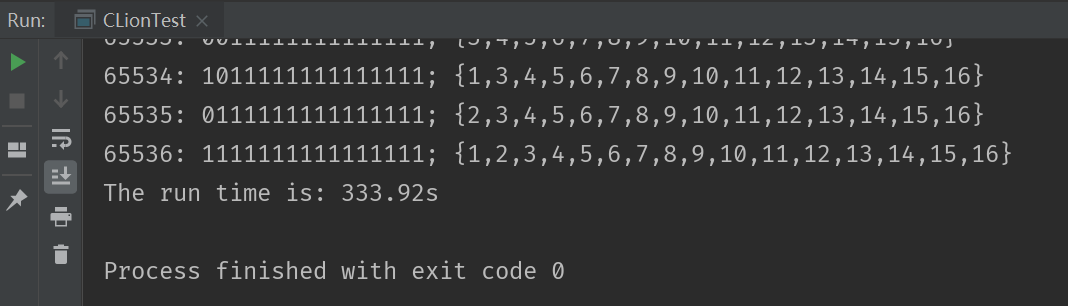
****

1. **在CAAIS中学习子集遍历算法，理解程序调用栈，以20%的交互运行n=4的子集遍历算法，并保存**

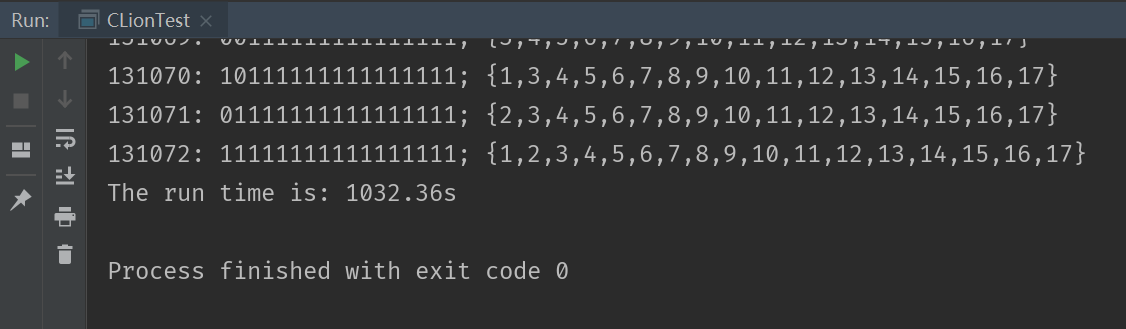


1. **在自己的机器上运行子集遍历算法，查看10分钟程序可以运行多大规模的子集遍历问题**

n = 16:



n = 17

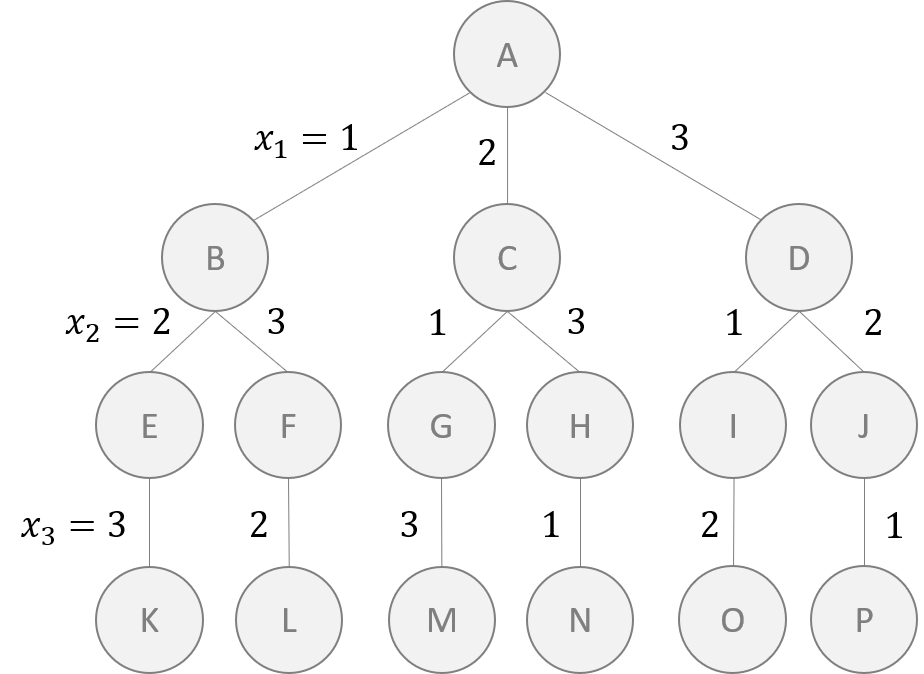


1. **给出TSP问题的最优化问题和判定性问题的定义**

最优化问题：给定n个顶点的无向或有向的带权图G（V，E），求一条包括各顶点一次且仅一次的最短环路。

判定性问题：给定n个顶点的无向或有向的带权图G（V，E），求一条从一个顶点出发经过每个顶点一次且仅一次且长度不大于的环路。

1. **画出n=3的排序树**

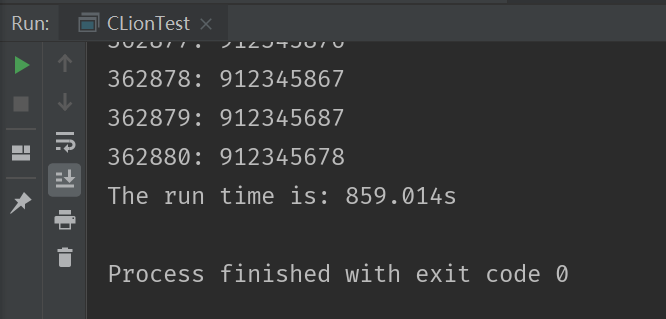
****

1. **在CAAIS中学习全排列遍历算法，以20%的交互运行您的全排列遍历算法，并将结果保存**



1. **在自己的机器上运行全排列遍历算法，查看30分钟可以运行多大规模的全排列遍历问题**

n=9:



1. **github地址**

<https://github.com/RuYunW/AlgorithmAnalysisAndDesign/tree/master/201711010202-王汝芸-2019-9-10>