Sch2-2：最佳调度问题

设有*n*个任务由*k*个可并行工作的机器来完成，完成任务*i*需要时间为。试设计一个算法找出完成这*n*个任务的最佳调度，使完成全部任务的时间最早。（要求给出调度方案）

完成全部任务的时间为运行时间最长的机器上运行的总时间，所有机器都是相同的。一个任务只能在一个机器上完成，且在完成之前不会被其他任务抢占。

三个测试样例

1. n=10, k=3, 任务的时间分别为47, 20, 28, 44, 21, 45, 30, 39, 28, 33
2. n=15, k=5, 任务的时间分别为98, 84, 50, 23, 32, 99, 22, 76, 72, 61, 81, 39, 76, 54, 37
3. n=19, k=8, 任务的时间分别为39, 39, 23, 45, 100, 69, 21, 81, 39, 55, 20, 86, 34, 53, 58, 99, 36, 45, 46

Tips

1. 设定剪枝策略，
2. 思考启发式的搜索方法，尽快找到最优解
3. 备份每个搜索过的结点