# Assignment #B: Dec Mock Exam大雪前一天

Updated 1649 GMT+8 Dec 5, 2024

2024 fall, Complied by <mark>祁黄奕 城市与环境学院</mark>

\*\*说明：\*\*

1）⽉考： AC6<mark>AC1</mark> 。考试题⽬都在“题库（包括计概、数算题目）”⾥⾯，按照数字题号能找到，可以重新提交。作业中提交⾃⼰最满意版本的代码和截图。

2）请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora https://typoraio.cn ，或者用word）。AC 或者没有AC，都请标上每个题目大致花费时间。

3）提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、"作业评论"区有上传的md或者doc附件。

4）如果不能在截止前提交作业，请写明原因。

## 1. 题目

### E22548: 机智的股民老张

http://cs101.openjudge.cn/practice/22548/

思路：找出一个整数数组中两个元素之间的最大差值，其中较大的元素必须位于较小元素之后。如果 i< minv，更新 minv 为当前元素 i。否则，计算 i 与 minv 之间的差值，并更新 maxv 为当前 maxv 和差值中的较大值。

代码：

```python

def maxvalue(a):  
 minv=a[0]  
 maxv=0  
 for i in a:  
 if i<minv:  
 minv=i  
 else:  
 maxv=max(maxv,i-minv)  
 return maxv  
lst=list(map(int,input().split()))  
print(maxvalue(lst))

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



### M28701: 炸鸡排

greedy, http://cs101.openjudge.cn/practice/28701/

思路：全部鸡排炸熟总时间除以锅的数量，得到平均每口锅的使用时间。耗时较长的鸡排视为占一口锅无法取下因此不断去除当前集合中超出“平均时间”的鸡排，锅的数量减1。

代码：

```python

n,k=map(int,input().split())  
t=list(map(int,input().split()))  
s=0.0  
for a in t:  
 s+=a  
t.sort(reverse=True)  
for a in t:  
 if a>s/k:  
 s-=a  
 k-=1  
print("%.3f"%(s/k))

```

代码运行截图 ==（至少包含有"Accepted"）==



### M20744: 土豪购物

dp, http://cs101.openjudge.cn/practice/20744/

思路：分两种情况，一种是拿出一个，另一种是不拿。事实上只要考虑两次贪心，

以第i个结尾的最大和，和以第i个开头的最大和。

代码：

```python

l=list(map(int,input().split(',')))  
n=len(l)  
def solve():  
 k=max(l)  
 if k<0:  
 return k  
 la1=[-float('inf')]\*n  
 la2=la1[:]  
 la1[0]=l[0]  
 la2[-1] = l[-1]  
 for i in range(1,n):  
 la1[i]=max(la1[i-1]+l[i],l[i])  
 la2[n-i-1]=max(la2[n-i]+l[n-i-1],l[n-i-1])  
 ans=max(la1)  
 for i in range(1,n-1):  
 ans=max(ans,la1[i-1]+la2[i+1])  
 return ans  
print(solve())

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



### T25561: 2022决战双十一

brute force, dfs, http://cs101.openjudge.cn/practice/25561/

思路：dfs暴力枚举法。

代码：

```python

result = float("inf")  
n, m = map(int, input().split())  
store\_prices = [input().split() for \_ in range(n)]  
you= [input().split() for \_ in range(m)]  
la=[0]\*m  
def dfs(i,sum1):  
 global result  
 if i==n:  
 jian=0  
 for i2 in range(m):  
 store\_j=0  
 for k in you[i2]:  
 a,b=map(int,k.split('-'))  
 if la[i2]>=a:  
 store\_j=max(store\_j,b)  
 jian+=store\_j  
 result=min(result,sum1-(sum1//300)\*50-jian)  
 return  
 for i1 in store\_prices[i]:  
 idx,p=map(int,i1.split(':'))  
 la[idx-1]+=p  
 dfs(i+1,sum1+p)  
 la[idx-1]-=p  
dfs(0,0)  
print(result)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



### T20741: 两座孤岛最短距离

dfs, bfs, http://cs101.openjudge.cn/practice/20741/

思路：dfs+bfs

代码：

```python

from collections import deque  
n=int(input())  
l=[list(input())for \_ in range(n)]  
la=[[False]\*n for \_ in range(n)]  
di=[(0,1),(0,-1),(1,0),(-1,0)]  
f=deque()  
def dfs(x,y):  
 for dx,dy in di:  
 nx,ny=x+dx,y+dy  
 if 0<=nx<n and 0<=ny<n and l[nx][ny]=='1':  
 l[nx][ny]=2  
 f.append((nx,ny))  
 dfs(nx,ny)  
def solve():  
 for i in range(n):  
 for j in range(n):  
 if l[i][j]=='1':  
 f.append((i,j))  
 l[i][j]=2  
 dfs(i,j)  
 return  
solve()  
def bfs():  
 ans=0  
 while f:  
 for \_ in range(len(f)):  
 x,y=f.popleft()  
 for dx, dy in di:  
 nx, ny = x + dx, y + dy  
 if 0 <= nx < n and 0 <= ny < n :  
 if l[nx][ny] == '1':  
 return ans  
 elif l[nx][ny]=='0':  
 l[nx][ny]=2  
 f.append((nx,ny))  
 ans+=1  
print(bfs())

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



### T28776: 国王游戏

greedy, http://cs101.openjudge.cn/practice/28776

思路：两个连续的数组之间谁排前面可以由一个函数决定，以此函数排序在以此更新最大值。

代码：

```python

n=int(input())  
a,b=map(int,input().split())  
l=[]  
for \_ in range(n):  
 x,y=map(int,input().split())  
 l.append((x,y))  
from functools import cmp\_to\_key  
def compare(a1, b1, a2, b2):  
 return (max(1 / b1, a1 / b2) >= max(1 / b2, a2 / b1)) - \  
 (max(1 / b1, a1 / b2) < max(1 / b2, a2 / b1))  
def compare\_wrapper(x, y):  
 return compare(x[0], x[1], y[0], y[1])  
l = sorted(l, key=cmp\_to\_key(compare\_wrapper))  
ans=a//l[0][1]  
for i in range(1,n):  
 a\*=l[i-1][0]  
 b\*=l[i-1][1]  
 ans=max(ans,a//l[i][1])  
print(ans)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



## 2. 学习总结和收获

月考难的无敌，本人菜的无敌。考场上第一题花了好多时间才勉强做出来，后面的几题复杂度也是十分之高，所以不参考题解写不出来。机考选择性放弃的策略更坚定了，希望最后期末能有简单题，求老师善良给过（别让孩子挂）呜呜呜……