

Assignment #C: bfs & dp

Updated 1436 GMT+8 Nov 25, 2025

2025 fall, Complied by 贺桢羽、心理与认知科学学院

说明：

- 1) 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。
- 2) 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。
- 3) 如果不能在截止前提交作业，请写明原因。

1. 题目

sy321迷宫最短路径

bfs, <https://sunnywhy.com/sfbj/8/2/321>

思路：bfs练习第一题。

代码：

```
from collections import deque
dx=[0,0,-1,1]
dy=[-1,1,0,0]
n,m=map(int,input().split())
ma=[]
for _ in range(n):
    ma.append(list(map(int,input().split())))
def bfs(sx,sy):
    q=deque()
    q.append((sx,sy))
    inq=set()
    prev = [[(-1, -1) for _ in range(m)] for _ in range(n)]
    inq.add((sx,sy))
    while q:
        x,y=q.popleft()
        if x==n-1 and y==m-1:
            return prev
        for i in range(4):
            tx=x+dx[i]
            ty=y+dy[i]
            if 0 <= tx < n and 0 <= ty < m and ma[tx][ty] == 0 and (tx, ty) not in inq:
                prev[tx][ty]=(x,y)
                inq.add((tx,ty))
```

```
        q.append((tx,ty))
    return None
prevc=bfs(0,0)
path=[]
end=(n-1,m-1)
while end!=(-1,-1):
    path.append(end)
    end=prevc[end[0]][end[1]]
path.reverse()
for pos in path:
    print(pos[0]+1,pos[1]+1)
```

代码运行截图 (至少包含有"Accepted")

```
1  from collections import deque
2  dx=[0,0,-1,1]
3  dy=[-1,1,0,0]
4  n,m=map(int,input().split())
5  ma=[]
6  for _ in range(n):
7      ma.append(list(map(int,input().split())))
8  def bfs(sx,sy):
9      q=deque()
10     q.append((sx,sy))
11     inq=set()
12     prev = [[(-1, -1) for _ in range(m)] for _ in range(n)]
13     inq.add((sx,sy))
14     while q:
15         x,y=q.popleft()
16         if x==n-1 and y==m-1:
17             return prev
18         for i in range(4):
19             tx=x+dx[i]
20             ty=y+dy[i]
21             if 0 <= tx < n and 0 <= ty < m and ma[tx][ty] == 0
22                 prev[tx][ty]=(x,y)
23                 inq.add((tx,ty))
24                 q.append((tx,ty))
25     return None
```

测试输入

提交结果

历史提交

完美通过

查看题解

100% 数据通过测试

详情

运行时长: 0 ms

收起面板

运行

提交

sy324多终点迷宫问题

bfs, <https://sunnywhy.com/sfbj/8/2/324>

思路：感觉套路性蛮强的。

代码：

```
from collections import deque
dx=[0,0,-1,1]
dy=[-1,1,0,0]
n,m=map(int,input().split())
ma=[]
for _ in range(n):
    ma.append(list(map(int,input().split())))
def bfs(sx,sy):
    ans = [[(-1) for i in range(m)] for _ in range(n)]
    q=deque()
    q.append((0,sx,sy))
    inq=set()
    inq.add((sx,sy))
    while q:
        step,x,y=q.popleft()
        for i in range(4):
            tx=x+dx[i]
            ty=y+dy[i]
            if 0 <= tx < n and 0 <= ty < m and ma[tx][ty] == 0 and (tx, ty) not in inq:
                ans[tx][ty]=step+1
                inq.add((tx,ty))
                q.append((step+1,tx,ty))
    return ans
ans_=bfs(0,0)
ans_[0][0]=0
for i in range(n):
    ans_[i]=list(map(str,ans_[i]))
print(" ".join(ans_[i]))
```

代码运行截图 (至少包含有"Accepted")

代码书写



Python ▾

```
1  from collections import deque
2  dx=[0,0,-1,1]
3  dy=[-1,1,0,0]
4  n,m=map(int,input().split())
5  ma=[]
6  for _ in range(n):
7      ma.append(list(map(int,input().split())))
8  def bfs(sx,sy):
9      ans = [[(-1) for i in range(m)] for _ in range(n)]
10     q=deque()
11     q.append((0,sx,sy))
12     inq=set()
13     inq.add((sx,sy))
14     while q:
15         step,x,y=q.popleft()
16         for i in range(4):
17             tx=x+dx[i]
18             ty=y+dy[i]
19             if 0 <= tx < n and 0 <= ty < m and ma[tx][ty] == 0 and (tx,
20                 ty) not in inq:
21                 ans[tx][ty]=step+1
22                 inq.add((tx,ty))
23                 q.append((step+1,tx,ty))
24     return ans
25 ans_=bfs(0,0)
26 ans_[0][0]=0
```

测试输入

提交结果

历史提交

查看题解

完美通过

100% 数据通过测试 [详情](#)

运行时长: 0 ms

收起面板

运行

提交

M02945: 拦截导弹

dp, greedy <http://cs101.openjudge.cn/pctbook/M02945>

思路: dp[i]指i为结尾时得到的最大拦截数量。

代码:

```

k=int(input())
li=list(map(int,input().split()))
dp=[1]*k
for i in range(1,k):
    for j in range(i):
        if li[i]<=li[j]:
            dp[i]=max(dp[i],dp[j]+1)
print(max(dp))

```

代码运行截图 (至少包含有"Accepted")

CS101 / 计算思维算法实践

题目 排名 状态 提问

#51002786 提交状态

查看 提交 统计 提问

状态: Accepted

基本信息

#:	51002786
题目:	M02945
提交人:	25n2500013720
内存:	3608kB
时间:	22ms
语言:	Python3
提交时间:	2025-11-25 22:10:19

源代码

```

k=int(input())
li=list(map(int,input().split()))
dp=[1]*k
for i in range(1,k):
    for j in range(i):
        if li[i]<=li[j]:
            dp[i]=max(dp[i],dp[j]+1)
print(max(dp))

```

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

189A. Cut Ribbon

brute force/dp, 1300, <https://codeforces.com/problemset/problem/189/A>

思路:

dp。要保证不越界，否则RE。

代码:

```

n,a,b,c=map(int,input().split())
dp=[-1008666]*(n+1)
li=[a,b,c]
li.sort()
a,b,c=li[0],li[1],li[2]
if a<=n:
    dp[a]=1
if b<=n:
    dp[b]=1
if c<=n:
    dp[c]=1
for i in range(a,n+1):
    if i-c>=0:
        dp[i]=max(dp[i-c]+1,dp[i-b]+1,dp[i-a]+1,dp[i])
    elif i-b>=0:
        dp[i] = max(dp[i - b] + 1, dp[i - a] + 1,dp[i])
    elif i-a>0:
        dp[i] =max(dp[i - a] +1,dp[i])
print(dp[n])

```

代码运行截图 (至少包含有"Accepted")

```
By hzy499, contest: Codeforces Round 119 (Div. 2), problem: (A) Cut Ribbon, Accepted, #, Copy
n, a, b, c=map(int, input().split())
dp=[-1008666]*(n+1)
li=[a, b, c]
li.sort()
a, b, c=li[0], li[1], li[2]
if a<=n:
    dp[a]=1
if b<=n:
    dp[b]=1
if c<=n:
    dp[c]=1
for i in range(a, n+1):
    if i-c>=0:
        dp[i]=max(dp[i-c]+1, dp[i-b]+1, dp[i-a]+1, dp[i])
    elif i-b>=0:
        dp[i] = max(dp[i - b] + 1, dp[i - a] + 1, dp[i])
    elif i-a>=0:
        dp[i] =max(dp[i - a] +1, dp[i])
print(dp[n])
```

→Judgement Protocol

Test: #1, time: 46 ms., memory: 8 KB, exit code: 0, checker exit code: 0, verdict: OK

Input
5 5 3 2
Output
2
Answer
2
Checker Log
ok 2

Test: #2, time: 62 ms., memory: 44 KB, exit code: 0, checker exit code: 0, verdict: OK

Input
7 5 5 2
Output
2
Answer
2
Checker Log

M01384: Piggy-Bank

dp, <http://cs101.openjudge.cn/practice/01384/>

思路：跟上一题很像。但是不知道用python3过不了，pypy3能过。

代码：

```
t=int(input())
for _ in range(t):
    e,f=map(int,input().split())
    n=int(input())
    money=[]
    dp=[1000000000 for _ in range(f-e+1)]
    for i in range(n):
        p,w=map(int,input().split())
        if w<=f-e:
            dp[w]=min(p,dp[w])
            money.append((w,p))
    money.sort()
    for x in range(1,f-e+1):
        for i in money:
            if x-i[0]>=1:
                dp[x]=min(dp[x-i[0]]+i[1],dp[x])
    if dp[f-e]>=100000000:
        print("This is impossible.")
```

```

else:
    print(f"The minimum amount of money in the piggy-bank is {dp[f-e]}.")

```

代码运行截图 (至少包含有"Accepted")

#51019292提交状态

状态: Accepted

基本信息

#: 51019292
题目: 01384
提交人: 25n2500013720
内存: 29472kB
时间: 393ms
语言: PyPy3
提交时间: 2025-11-26 20:11:56

源代码

```

t=int(input())
for _ in range(t):
    e,f=map(int,input().split())
    n=int(input())
    money=[]
    dp=[100000000 for _ in range(f-e+1)]
    for i in range(n):
        p,w=map(int,input().split())
        if w<=f-e:
            dp[w]=min(p,dp[w])
            money.append((w,p))
    money.sort()
    for x in range(1,f-e+1):
        for i in money:
            if x-i[0]>=1:
                dp[x]=min(dp[x-i[0]]+i[1],dp[x])
    if dp[f-e]>=100000000:
        print("This is impossible.")
    else:
        print(f"The minimum amount of money in the piggy-bank is {dp[f-e]}.")

```

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

M02766: 最大子矩阵

dp, kadane, <http://cs101.openjudge.cn/pctbook/M02766>

思路:

学习了kadane算法。

代码:

```

from collections import deque
n=int(input())
ma_temp=deque()
answer=float("-inf")
while 1:
    try:
        li=list(map(int,input().split()))
        ma_temp.extend(li)
    except EOFError:
        break
ma=[]
for _ in range(n):
    li=[]
    for _ in range(n):
        li.append(ma_temp.popleft())
    ma.append(li)
for x in range(n):

```

```

temp=[0 for _ in range(n)]
for y in range(x,n):
    for i in range(n):
        temp[i]+=ma[i][y]
ans=now=temp[0]
for s in range(1,n):
    now=max(temp[s],now+temp[s])
    ans=max(now,ans)
answer=max(answer,ans)
print(answer)

```

代码运行截图 (至少包含有"Accepted")

#51020299提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

from collections import deque
n=int(input())
ma_temp=deque()
answer=float("-inf")
while 1:
    try:
        li=list(map(int,input().split()))
        ma_temp.extend(li)
    except EOFError:
        break
ma=[]
for _ in range(n):
    li=[]
    for _ in range(n):
        li.append(ma_temp.popleft())
    ma.append(li)
for x in range(n):
    temp=[0 for _ in range(n)]
    for y in range(x,n):
        for i in range(n):
            temp[i]+=ma[i][y]
        ans=now=temp[0]
        for s in range(1,n):
            now=max(temp[s],now+temp[s])
            ans=max(now,ans)
        answer=max(answer,ans)
print(answer)

```

基本信息

#: 51020299
 题目: M02766
 提交人: 25n2500013720
 内存: 28816kB
 时间: 136ms
 语言: PyPy3
 提交时间: 2025-11-26 20:47:02

2. 学习总结和收获

如果作业题目简单，有否额外练习题目，比如：OJ“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。

第一次接触bfs，感觉还不错，就是跟着模板写代码。这周继续进行了dp的巩固，拦截导弹这个题感觉比较难想，又积累了一些dp的知识。第四第五题比较相似。第六题积累了kadane算法。现在洛谷练习dfs的题目，dfs掌握得最差。