"""

OJ23563: 多项式时间复杂度

http://cs101.openjudge.cn/practice/23563/

n, m = map(int, input().split())  
lst = [[0] \* (m + 2)]  
for i in range(n):  
 lst.append([0] + list(map(int, input().split())) + [0])  
lst.append([0] \* (m + 2))  
ans = [[]] \* n  
for i in range(1, n + 1):  
 for j in range(1, m + 1):  
 sv = lst[i - 1][j - 1] + lst[i][j - 1] + lst[i - 1][j] + lst[i - 1][j + 1] + lst[i][j + 1] + lst[i + 1][  
 j - 1] + lst[i + 1][j] + lst[i + 1][j + 1]  
 if j-m:  
 if lst[i][j]:  
 print(int(sv == 2 or sv == 3), end=" ")  
 else:  
 print(int(sv == 3), end=" ")  
 else:  
 if lst[i][j]:  
 print(int(sv == 2 or sv == 3))  
 else:  
 print(int(sv == 3))

"""

"""

OJ23566: 决战双十一

http://cs101.openjudge.cn/practice/23566/

n, m, k = map(int, input().split())  
lst = [[False] \* (m + 2) for i in range(n + 2)]  
for i in range(k):  
 x, y = map(int, input().split())  
 lst[x][y] = True  
 if lst[x - 1][y - 1] and lst[x][y - 1] and lst[x - 1][y] or \  
 lst[x - 1][y + 1] and lst[x][y + 1] and lst[x - 1][y] or \  
 lst[x + 1][y - 1] and lst[x][y - 1] and lst[x + 1][y] or \  
 lst[x + 1][y + 1] and lst[x][y + 1] and lst[x + 1][y]:  
 print(i + 1)  
 exit()  
 elif i == k - 1:  
 print(0)

"""

"""

OJ19948: 因材施教(greedy)

http://cs101.openjudge.cn/practice/19948/

n, m = map(int, input().split())  
ai = list(map(int, input().split()))  
dct, ans = {}, []  
for i in range(n - 1, -1, -1):  
 dct[ai[i]] = True  
 ans.append(len(dct))  
for i in range(m):  
 print(ans[n-int(input())])

"""

"""

OJ25301: 生日相同

http://cs101.openjudge.cn/practice/25301/

n = int(input())  
xi = [int(i) for i in input().split()]  
xi.sort()  
q = int(input())  
for i in range(q):  
 mi = int(input())  
 if mi >= xi[n - 1]:  
 print(n)  
 else:  
 d, u = 0, n - 1  
 while d - u:  
 m = (u + d) // 2  
 if xi[m] > mi:  
 u = m  
 else:  
 d = m+1  
 print(u)

"""

"""

OJ25302: 最大并发量

http://cs101.openjudge.cn/practice/25302/

m, n = map(int, input().split())  
num, position, lst, ans = {}, {}, [], 0  
for i in range(m):  
 tmp = tuple(map(int, input().split()))  
 for j in range(n):  
 num[tmp[j]] = (i, j)  
for i in range(m \* n):  
 tmp = tuple(map(int, input().split()))  
 position[num[i]] = tmp  
 lst.append(sum(tmp))  
for i in range(m):  
 for j in range(n):  
 if (i and position[(i, j)] == position[(i - 1, j)]) or \  
 (i - m + 1 and position[(i, j)] == position[(i + 1, j)]) or \  
 (j and position[(i, j)] == position[(i, j - 1)]) or \  
 (j - n + 1 and position[(i, j)] == position[(i, j + 1)]):  
 ans += 1  
print(ans, end=" ")  
lst.sort(reverse=True)  
ans = m \* n \* 2 // 5  
while lst[ans] == lst[ans-1] and ans > 0:  
 ans -= 1  
print(ans)

"""

"""

CF1749C: Number Game, binary search/data structure/games/greedy/implementation, 1400

https://codeforces.com/problemset/problem/1749/C

n, a, b, c = map(int, input().split())  
mn = min(a, b, c)  
mx = max(a, b, c)  
md = a + b + c - mn - mx  
ans = 0  
if mn == 1:  
 print(n)  
 exit()  
for i in range(n // mn, -1, -1):  
 for j in range((n - mn \* i) // md, -1, -1):  
 for k in range((n - mn \* i - md \* j) // mx, -1, -1):  
 if n == mn \* i + md \* j + mx \* k:  
 ans = max(ans, i + j + k)  
 if i + j + k < ans - 2:  
 print(ans)  
 exit()  
print(ans)

"""