1

23

456

78910

456

23

1

def floyd(rowcount=5):

rows = [[1]]

while len(rows) < rowcount:

n = rows[-1][-1] + 1

rows.append(list(range(n, n + len(rows[-1]) + 1)))

return rows

def pfloyd(rows=[[1], [2, 3], [4, 5, 6], [7, 8, 9, 10]]):

colspace = [len(str(n)) for n in rows[-1]]

for row in rows:

print( ' '.join('%\*i' % space\_n for space\_n in zip(colspace, row)))

def pascal(n):

"""Prints out n rows of Pascal's triangle.

It returns False for failure and True for success."""

row = [1]

k = [0]

for x in range(max(n,0)):

print row

row=[l+r for l,r in zip(row+k,k+row)]

return n>=1