Homework 4

1. Consider an economy where the price of the risky security (“stock”) can go up 12% or down 8% per period. The interest rate is 5% per period. Compute the price of another security (“derivative”) that pays $1000 if the stock goes down for five consecutive periods and costs $40 if the stock goes up for five consecutive periods (its payoff is zero in all other scenarios).

```
from math import factorial as fact
import numpy as np
r = 0.05
u = 1 + 0.12
d = 1 + -0.08
p = ( 1 + r - d )/(u - d)
q = ( u - (1 + r))/ ( u - d)
n = 5
c = 1/(1 + r)**n *sum( [fact(5)/fact(5) * p**5 * -40, fact(5)/fact(5) * q
**5 * 10000])
```

set the pay off equal to 1000 and - 40. We can sum the expected payoff and discount them to today to get the price of the security in the no arbitrage world.
solve price equal to 43.367036507858515

2. Consider the following trinomial economy: the price of the risky stock is $100. After one period, the price of the stock can go up to $120, stay at $100 or go down to $90. The price of an European call with strike price $105 is $5. The price of an European call with strike price $95 is $10. What would be the price of a security that would pay $1 after one period regardless of the state?

we can use stock , call option 1 and call option2 to replicate the security. And for no arbitrage, the price of security should be the weighted average of those three.

```
e1 = sym.Eq(d * 120 + c1 * 15 + c2 * 25 ,1)
e2 = sym.Eq(d * 100 + c1 * 0  + c2 * 5 , 1 )
e3 = sym.Eq(d * 90 + c1*0 + c2 * 0 , 1 )
weight = sym.solve([e1 ,e2, e3],[d,c1,c2]).values()
weight = np.array(list(weight))
print(weight)
price = float(sum(weight *np.array([100, 5,10])) )
print(price)
rate = 1/float(sum(weight *np.array([100, 5,10])) )
print(rate)
```

$$d*120+c1*15+c2*25=1$$
$$d*100+c1*0+c2*5=1$$
$$d*-10+c1*0+c2*0=1$$

solve for d, c1 ,c2, we have $[1/902/135 - 1/45]$
the price is $sum([1/90, 2/135, -1/45] * [100, 5, 10]) = 0.9629629629629629$
And the risk free rate would be $1/0.963 - 1 = 0.038461535$

3. Consider the following two-period setting. The price of a stock is $50. Interest rate per period is 2%. After one period the price of the stock can go up to $55 or drop to $47 and it will pay (in both cases) a dividend of $3. If it goes up the first period the second period can go up to $57 or down to $48. If it goes down the first period, the second period can go up to $48 or down to $41. a) Compute the price of an American put option with strike price X = 45 that matures at the end of the second period. b) Compute the price of an American call option with strike price X = 45 that matures at the end of the second period.

A

lets calculate the price of Put_u and Put_d
Notice we do not minus the dividend when we calculate the probability since it is only one time dividend (desecrate)

```
dividend = 3
s = 50
u = 55 -3
d = 47 -3
uu = 57
ud = 48
du = 48
dd = 41
r = 0.02
k = 45
cuu = max(-(uu - k), 0 )
cud = max(-(ud - k), 0 )
cdu = max(-(du - k), 0 )
cdd = max(-(dd - k), 0 )
puu = ( 1 + r - ud/u )/(uu/u - ud/u)
pud = (uu/u - (1 + r))/  (uu/u - ud/u)
pdu = ( 1 + r - dd/d)/(du/d - dd/d)
pdd = (du/d - (1 + r))/ (du/d - dd/d)
pu = ( 1 + r - (d+3)/s )/((u+3)/s - (d+3)/s)
pd = ((u+3)/s - (1 + r))/  ((u+3)/s - (d+3)/s)
put_u =1/(1+r)*( puu * cuu + pud * cud)
put_d = 1/(1+r)*( pdu * cdu + pdd * cdd)
print(put_u , put_d)
p =1/(1+r) *( pu*put_u + pd * 2)
```

solve probability of up twice $P_{uu} = (1 + r - S_{ud}/S_u)/(S_{uu}/S_u - S_{ud}/S_u) = 0.56$
and probability of up then down $P_{ud} = (S_{uu}/S_u - (1 + r))/(S_{uu}/S_u - S_{ud}/S_u) = 0.44$
Notice that the price of S_u is the price after dividend.
Then the price of put option at S_u is $Put_u = 1/(1 + r) * (P_{uu} * Put_{uu} + P_{ud} * Put_{ud}) = 0$
same process the $Put_d = 1.7478991596638633$
notice payoff at S_u is 0, no early exercise. However, the payoff at S_d is

$47 - 45 = 2 > 1.747899$. We should exercise.
finally, $Put = 1/(1+r) * (P_u * Put_u + p_d * 2) = 0.9803$

B

```
dividend = 3
u = 55 -3
d = 47 -3
uu = 57
ud = 48
du = 48
dd = 41
r = 0.02
k = 45
cuu = max(uu - k, 0 )
cud = max(ud - k, 0 )
cdu = max(du - k, 0 )
cdd = max(dd - k, 0 )
puu = ( 1 + r - ud/u )/(uu/u - ud/u)
pud = (uu/u - (1 + r))/  (uu/u - ud/u)
pdu = ( 1 + r - dd/d)/(du/d - dd/d)
pdd = (du/d - (1 + r))/ (du/d - dd/d)
pu = ( 1 + r - (d+3)/s )/((u+3)/s - (d+3)/s)
pd = ((u+3)/s - (1 + r))/  ((u+3)/s - (d+3)/s)
cu =1/(1+r)*( puu * cuu + pud * cud)
cd = 1/(1+r)*( pdu * cdu + pdd * cdd)
print(cu , cd)
c =1/(1+r) *( pu*10 + pd * 2)
```

same process with difference payoff for call option
$C_u = 1/(1+r) * (P_{uu} * C_{uu} + P_{ud} * C_{ud}) = 7.8823 > (55 - 45)$
$C_d = 1/(1+r) * (P_{du} * C_{du} + P_{dd} * C_{dd}) = 1.6302 > (47 - 45)$

early exercise in both situation
thus, $C = 1/(1+r) * (P_u * 10 + P_d * 2) = 5.88235294117647$