PROBLEM 1

```
BSM vs Finite Difference Greeks Comparison:
Call Option Greeks (BSM): {'delta': 0.083011070896926869, 'gamma': 0.016830979206204362, 'vega': 6.938710929513443, 'theta': -8.196615884194502, 'rho': 1.1025939156368187}
Call Option Greeks (Finite Difference): {'delta': 0.08297130338341674, 'gamma': 0.016824763804379472, 'vega': 6.938710350752331, 'theta': -8.126520224163158, 'rho': 1.102593926054496}
Put Option Greeks (BSM): {'delta': -0.9169889291037313, 'gamma': 0.016830979206204362, 'vega': 6.938710929513443, 'theta': -1.2110094733534407, 'rho': -13.758003122735788}
Put Option Greeks (Finite Difference): {'delta': -0.9165496332741441, 'gamma': 0.016834178495628294, 'vega': 6.938710350681276, 'theta': -1.9409893427280167, 'rho': -13.758003112513961

Option Prices Comparison (BSM vs Binomial):
Call Price (BSM): 0.3358
Put Price (BSM): 13.7454
Call Price (Binomial, without dividend): 0.3418
Put Price (Binomial, without dividend): 165.0000
Call Price (Binomial, with dividend): 0.3418
Put Price (Binomial, with dividend): 165.3200
Call Option Greeks (Binomial Tree): {'delta': 0.07963234403440822, 'gamma': -1.4432899320127035e-07, 'vega': 6.71240557054309, 'theta': -7.947810146471112, 'rho': 1.113612033041278}
Put Option Greeks (Binomial Tree): {'delta': 0.0, 'gamma': 0.0, 'vega': 0.0, 'theta': 0.0, 'rho': 0.0}
```

Problem2

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
: 0
: 0.4672694327936942
       Portfolio   Value       VaR        ES
0       Straddle    6.80 -0.081780 -0.000022
1       Straddle    4.85 -0.054277  0.000008
2        SynLong    6.80 -0.081780 -0.000006
3        SynLong   -4.85  0.082533       NaN
4     CallSpread    6.80 -0.081780 -0.000005
5     CallSpread   -2.21  0.045298  0.002235
6      PutSpread    4.85 -0.054277 -0.000011
7      PutSpread   -1.84  0.040079  0.001764
8          Stock  151.03 -2.116015  0.000444
9           Call    6.80 -0.081780  0.000021
10           Put    4.85 -0.054277  0.000010
11   CoveredCall  151.03 -2.116015  0.000208
12   CoveredCall   -4.05  0.071250       NaN
13  ProtectedPut  151.03 -2.116015  0.000583
14  ProtectedPut    3.01 -0.028325 -0.000018
<ipython-input-32-e74fafe7dcab>:78: RuntimeWarning: divide by zero encountered in scalar divide
  d1 = (np.log(S / K) + (r + 0.5 * sigma ** 2) * T) / (sigma * np.sqrt(T))
/usr/local/lib/python3.10/dist-packages/numpy/core/fromnumeric.py:3504: RuntimeWarning: Mean of empty slice.
  return _methods._mean(a, axis=axis, dtype=dtype,
/usr/local/lib/python3.10/dist-packages/numpy/core/_methods.py:129: RuntimeWarning: invalid value encountered in scalar divide
  ret = ret.dtype.type(ret / rcount)
```

problem 3

```
Expected Annual Returns:
Empty DataFrame
Columns: [Stock, Expected Annual Return]
Index: []
Annual Covariance Matrix:
          SPY      AAPL      MSFT      AMZN      TSLA     GOOGL      GOOG  \
SPY   0.015333  0.015632  0.016999  0.024130  0.033242  0.019676  0.019605
AAPL  0.015632  0.050501  0.021561  0.024202  0.046972  0.025260  0.025196
MSFT  0.016999  0.021561  0.039604  0.035189  0.029237  0.029757  0.029453
AMZN  0.024130  0.024202  0.035189  0.081874  0.046419  0.045175  0.044725
TSLA  0.033242  0.046972  0.029237  0.046419  0.293876  0.031925  0.031884
...        ...       ...       ...       ...       ...       ...       ...
LMT   0.000404  0.001382 -0.002160 -0.002250 -0.007311 -0.006084 -0.005997
SYK   0.010617  0.007993  0.010191  0.015846  0.016553  0.008585  0.008626
GM    0.015345  0.005763  0.006278  0.009179  0.036871  0.005554  0.005741
TFC   0.017508  0.009470  0.007391  0.013459  0.048347  0.009384  0.009847
TJX   0.010103  0.008573  0.010275  0.014532  0.017376  0.011252  0.011385

          META      NVDA     BRK-B  ...       PNC      MDLZ        MO  \
SPY   0.024700  0.039266  0.007982  ...  0.013525  0.004436  0.005375
AAPL  0.020870  0.033939  0.006499  ...  0.005322  0.002506  0.005324
MSFT  0.041722  0.045418  0.006653  ...  0.001614  0.002501  0.005841
AMZN  0.064354  0.067779  0.008978  ...  0.008737  0.000145  0.002156
TSLA  0.033953  0.071756  0.009049  ...  0.038982  0.012639  0.010996
...        ...       ...       ...  ...       ...       ...       ...
LMT  -0.008482 -0.018103  0.003067  ...  0.004058  0.006682  0.007193
SYK   0.016414  0.019059  0.006795  ...  0.006392  0.007807  0.006941
GM    0.005321  0.014304  0.014117  ...  0.041617  0.004688  0.009204
TFC   0.010727  0.011974  0.015694  ...  0.058672  0.008318  0.009365
TJX   0.014315  0.016990  0.007174  ...  0.010092  0.007098  0.008099

           ADI      GILD       LMT       SYK        GM       TFC       TJX
SPY   0.023697  0.003047  0.000404  0.010617  0.015345  0.017508  0.010103
AAPL  0.021843 -0.001960  0.001382  0.007993  0.005763  0.009470  0.008573
MSFT  0.020152  0.001182 -0.002160  0.010191  0.006278  0.007391  0.010275
AMZN  0.029404  0.002144 -0.002250  0.015846  0.009179  0.013459  0.014532
TSLA  0.064953 -0.001760 -0.007311  0.016553  0.036871  0.048347  0.017376
...        ...       ...       ...       ...       ...       ...       ...
LMT  -0.005877  0.006598  0.030799  0.001804  0.000159  0.003187  0.003422
SYK   0.018610  0.001829  0.001804  0.045443  0.014758  0.009800  0.008725
GM    0.035177  0.013321  0.000159  0.014758  0.093479  0.045567  0.011920
TFC   0.030777  0.018943  0.003187  0.009800  0.045567  0.082699  0.009707
TJX   0.017552  0.006485  0.003422  0.008725  0.011920  0.009707  0.029437

[100 rows x 100 columns]
num_stocks: 0
initial_weights: []
Bounds: []
Optimization failed, error message: not enough values to unpack (expected 2, got 0)
```