Problem 1

- *Current Stock Price $165*
- *Strike Price $165*
- *Current Date 03/13/2022*
- *Options Expiration Date 04/15/2022*
- *Risk Free Rate of 4.25%*
- *Continuously Compounding Coupon of 0.53%*

Implement the closed form greeks for GBSM. Implement a finite difference derivative calculation.
Compare the values between the two methods for both a call and a put.
Implement the binomial tree valuation for American options with and without discrete dividends.
Assume
the stock above:

- *Pays dividend on 4/11/2022 of $0.88*

Calculate the value of the call and the put. Calculate the Greeks of each.
What is the sensitivity of the put and call to a change in the dividend amount?

1. First we have the formula for the Greeks Black Scholes Method implement a function that have option of "Put" or "Call", with default Call option. The function will return the values of Greeks in sequence. Call both Call option and Put option and show it in a Data Frame.
2. Write the function that calculate Greeks using finite difference.

	Call Closed-form	Call Finite Diff.	Put Closed-form	Put Finite Diff.
Delta	0.534009	0.533743	-0.465512	-0.465778
Gamma	0.053818	0.039949	0.053818	0.039949
Vega	19.710180	19.710095	19.710180	19.710095
Theta	-26.233902	-25.067507	-32.429640	-18.955581
Rho	7.583586	7.583554	-7.277011	-7.277045

3. Write a binomial tree function that take in account of dividend and dividend date, as well as Greeks calculation function.
4. After input the dividend and date, we get the result:
5.

```
American Call Option without Dividends: 4.262
American Put Option without Dividends: 3.681
American Call Option with Dividends: 3.812
American Put Option with Dividends: 4.116
```

```
Call Option Greeks:
       Call without div  Call with div
Delta          0.536898       0.500120
Gamma          0.038292       0.045113
Theta        -25.932238     -25.884008
Vega          19.657366      19.821456
Rho            7.550520       7.066599

Put Option Greeks:
       Put without div  Put with div
Delta        -0.471247     -0.508301
Gamma         0.039587      0.046210
Theta       -19.577328    -19.518492
Vega         19.621387     19.702896
Rho          -5.868226     -6.173442
```

6. For sensitivity, we tried to calculate the price changes per dividend to determine the sensitivity of put and call options

```
Delta Call with Dividend: -0.5115
Delta Put with Dividend: 0.4945
```

Conclusion:

It shows that The Call option price have a moderate negative relationship with dividend and Put option price have a moderate positive relationship with dividend. In other word, more dividend means a decrease in price of call option, and an increase in price of put option.

Problem 2

Using the options portfolios from Problem3 last week (named problem2.csv in this week's repo) and assuming :

- *American Options*
- *Current Date 03/03/2023*
- *Current AAPL price is 151.03*
- *Risk Free Rate of 4.25%*
- *Dividend Payment of $1.00 on 3/15/2023*

Using DailyPrices.csv. Fit a Normal distribution to AAPL returns – assume 0 mean return. Simulate AAPL
returns 10 days ahead and apply those returns to the current AAPL price (above). Calculate Mean, VaR and ES.
Calculate VaR and ES using Delta-Normal.
Present all VaR and ES values a $ loss, not percentages.
Compare these results to last week's results.

1. Call the function in the risk_mgmt package that add Time to maturity and implied volatility to the data frame
2. Read price of AAPL and use package function to calculate the log_returns and then minus mean of it.
3. Use norm in scipy.stats library to fit the data we get from last step and get mu and standard deviation.
4. Do a 1000 times simulation of 10 days ahead prices.
5. Calculate the portfolio values of 10 days ahead and current portfolio values to find the PnL of the portfolio. Finally call package that calculate ES and VaR:

	Mean	VaR	ES
Straddle	1.804015	1.378254	1.38726
SynLong	0.047086	16.852753	20.732196
CallSpread	-0.083755	3.958245	4.229466
PutSpread	0.326626	2.700865	2.839366
Stock	0.25291	16.605965	20.463731
Call	0.925551	6.115365	6.415636
Put	0.878464	4.46001	4.637761
CoveredCall	-0.803667	12.760137	16.515701
ProtectedPut	1.000607	8.210134	8.784374

6. Write a function that use Delta Normal to calculate VaR and ES:

	VaR	ES
Straddle	2.251019	2.773958
SynLong	16.579459	20.431067
CallSpread	4.687565	5.776543
PutSpread	4.31774	5.427035
Stock	16.605965	20.463731
Call	9.415239	11.602513
Put	8.093689	10.173085
CoveredCall	9.536185	11.751556
ProtectedPut	11.576971	14.266441

7. Comparing the result to we did last week, we can find that the method using Delta Normal is the most conservative one. The VaR and ES are both higher than the other two methods. And the values we did this week (Normal model fitted) is slightly larger than the values in last week (AR(1) model fitted).

	Mean	VaR	ES
Straddle	1.482665	1.368381	1.385275
SynLong	-0.138553	16.847177	20.226246
CallSpread	-0.116082	3.95765	4.204266
PutSpread	0.314248	2.564907	2.758005
Stock	0.064347	16.600427	19.960067
Call	0.672056	6.114688	6.388221
Put	0.810609	4.280293	4.531878
CoveredCall	-0.745402	12.754859	16.020504
ProtectedPut	0.745722	8.208921	8.729172

Problem 3

Use the Fama French 3 factor return time series (F-F_Research_Data_Factors_daily.CSV) as well as the Carhart Momentum time series (F-F_Momentum_Factor_daily.CSV) to fit a 4 factor model to the following
stocks.
AAPL FB UNH MA
MSFT NVDA HD PFE
AMZN BRK-B PG XOM
TSLA JPM V DIS
GOOGL JNJ BAC CSCO
Fama stores values as percentages, you will need to divide by 100 (or multiply the stock returns by 100)
to get like units.
Based on the past 10 years of factor returns, find the expected annual return of each stock.
Construct an annual covariance matrix for the 10 stocks.

1. We loaded the Fama-French, momentum, and daily prices datasets using Pandas library.
2. Then, we preprocessed the datasets by converting the date format and calculating stock returns using the 'return_calculate' function. We selected 20 stocks for the analysis and calculated their returns. We also subtracted the risk-free rate from the stock returns to calculate the excess returns.
3. For each selected stock, we fit a OLS model to get the betas and alphas for calculating the annually expected returns:

AAPL	-0.227886	PG	0.008929
META	-0.485602	XOM	0.475491
UNH	0.171017	TSLA	-0.767201
MA	-0.027627	JPM	-0.073974
MSFT	-0.161588	V	-0.041539
NVDA	-0.373268	DIS	-0.556849
HD	-0.043308	GOOGL	-0.413719
PFE	0.095651	JNJ	0.118909
AMZN	-0.577929	BAC	-0.3292
BRK-B	0.006792	CSCO	-0.050649

4. Calculate the 12 stocks covariance matrix based on the returns.
5. Write a function that returns negative Sharpe ratio. Use minimize from scipy.optimize to minimize negative Sharpe ratio, which is the same as maximize the Sharpe ratio as well as weights:

Stock	AAPL	META	UNH	MA	MSFT	NVDA	HD	PFE	AMZN	BRK-B	PG	XOM	TSLA	JPM	V	DIS	GOOGL	JNJ	BAC	CSCO
Weight	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.676	0.0	0.0	0.0	0.0	0.0	0.324	0.0	0.0

Conclusion:

With our optimize model, we would hold 0.676 of XOM and 0.324 JNJ with a Sharpe ratio = 1.285 based on the historical data and expected returns