BreakPointCaclulator

1. int sharsOutstanding
2. double issuePrice
3. double liquidationPreference
4. boolean participating
5. double dividend
6. int senority
7. int breakPoint

1. int breakPoint(sharesOutstanding, senority,issuePrice liquidationPreference): return breakpoint
2. int breakPoint(sharesOutstanding, issuePricesenority, liquidationPreference,participating): return int breakPoint

OptionPricingCalculator

double d1
double d2
double nD1
double nD2
double strikePrice
int equityValue
double volatility
double timeToLiquidity
double rFRate
double dividendYield
int incrementalValue
int breakpoint
double optionPrice
list optionPriceList
list incrementalValueList

getND1(equityValue,volatility, timeToLiquidity, rFrate, dividendYield):return double d1,nD1
etND2(equityValue,volatility, timeToLiquidity, rFrate, dividendYield):return double d2,nD2
getOptionPrice(nD1,nD2,equityValue,volatility, timeToLiquidity, rFrate, dividendYield) return int optionPrice
getIncrementalValue([optionPrice]):return list incrementalValue

AllocationCalculator

1.double allocationPercentage
2. int sharesOustanding
3. int issuePrice
4.double liquidationPreference
5. list incrementalValueList
6.list allocationPercentageList
7. list optionValueList
8.list sharesOutstanding

getAllocationPercentage(sharesOustandng) return double allocationPercentage
getAllocationPercentage(sharesOustandng, int issuePrice) return double allocationPercentage
getAllocationPercentage(sharesOustandng, int issuePrice) return double allocationPercentage
get optionValue(list allocationPercentageList, incrementalValueList) return optionValueList

FairValue

1. list optionValueList
2. list sharesOutstandingList
3. int fairValue
4. list fairValueList

getFairValueList(optionValueList,sharesOutstanding, timeToLiquidity) return fairValueList