Input Parameters

- Original ranking `existingResults`
- Diversification depth (`k=30`)
- `w`, weighting for diversification scoring component

Output

- Manipulated set of results, diversified to `k`

Helper Functions

- `getEntities(x,y,z)` Returns an array results `x`, returns an array of entities present in documents in array range `y` to `z`
- `getLength(x)` Returns the length of array `x`
- `getUnseenEntities(x,y)` Returns entities in document `x` that have not yet been observed in ranked document array `y`
- `sortByScore(x)` Sorts document array `x` by `score` in descending order
- `<array>.pop()` Removes the top entry from an array, returning the popped value

```
SET entities TO []
SET newRankings TO []
SET i TO 1

# Take the top result from the baseline results, popping results
SET newRankings[0] TO existingResults.pop()

WHILE i <= k DO
    SET entities TO getEntities(existingResults, 0, i-1)
    SET j TO 0

    WHILE j <= getLength(existingResults) DO
        SET newEntityCount TO
                getUnseenEntities(document, existingResults)
        SET existingResults[j].score TO score + (w·newEntityCount)
        SET j TO j + 1
    END WHILE

    sortByScore(existingResults)
    # existingResults now re-ordered, move top result to new list
    SET newRankings[i] TO existingResults.pop()
    SET i TO i + 1
END WHILE
```