**Open Word :** For Project 1 I am considering it as Knapsack problem where we have 50 songs with different duration and we need to select some songs for the next album there total duration should be 30 minutes (1800000 ms.) So which means we have a bag that is having capacity of 1800000 and we need to fill it with different songs so it should be filled.

**Approach and assumption:**

I am solving this problem using Bottom Up approach there I will create a matrix of 50\*1800000.

When I am creating this matrix and trying to use it I am getting out of memory issues. So to optimize the space issue I decided to reduce the space.

I check the average duration of song which is around 188258 so the album will have approx. 10 songs of such duration. That is why I decided to reduce the time of each songs by dividing by 10 and total duration as well by divided by 10.

After doing this I resolved the out of memory issue and improve the execution time of the problem.

**Explanation of code**

Divide the total duration by 10.

Print the memory footprint

Create the Keep matrix of number of songs and total duration (180000)

Create temporary matrix Called V of number of songs and total duration (180000)

Create solution array of all songs it will have value 1 for selected sngs.

Divide the duration of each songs by 10.

Fill the matrix V by 0.

Now run a loop from 1 to totals songs

Run another loop from 0 to total duration

If duration of current songs is less than total duration

Get the max of previous song duration + duration from matrix V and duration from previous row of V

Update the value of matrix of keep by 1

Else

Update the value of V by previous row

Update the value of keep by 0

Now fill the solution array based on keep matrix where value is 1

Reduce the total duration by current song duration.

**Output:**

Initial Memory footprint

Initial memory: 0.13 GB

Used heap memory: 0.00 GB

Max heap memory: 1.76 GB

Committed memory: 0.12 GB

49 --> Song included

48 --> Song included

47 --> Song included

46 --> Song included

45 --> Song included

44 --> Song included

43 --> Song included

42 --> Song included

41 --> Song included

40 --> Song included

39 --> Song included

37 --> Song included

35 --> Song included

After execution memory usage

Initial memory: 0.13 GB

Used heap memory: 0.17 GB

Max heap memory: 1.76 GB

Committed memory: 0.31 GB

**Total duration:** 1788170

**Obj: 533**

**Gap: 11830 ms.**