

HOMework 4
COMP3121 - ALGORITHM DESIGN

QUESTION 3

PUNEETH KAMBHAMPATI
z5164647

SOLUTION:

```
# days is a 2D MATRIX that contains the enjoyment scores
# for all activities on each day
# example element: [activity1, activity2, activity3]
# [
# [3,1,2], → Day 1
# [5,2,6], → Day 2
# [4,2,8] → Day 3
# ]

class Q3:
    def maxEnjoyment(days):
        # If we are at the first day the return
        # the enjoyment scores for that day
        if len(days) == 1:
            return days[0]

        # FunToday has the enjoyment scores for today
        FunToday = days[-1]
        # FunTillToday recursively finds the maximum
        # enjoyment we could have had till the current day
        FunTillToday = Q3.maxEnjoyment(days[:-1])

        # store the max enjoyment scores for today in maxFun
        maxFun = []
        # For EACH activity find the maximum fun by
        # summing today's score with the most optimal choice from yesterday\
        temp = 0
        for activityNum in range(3):
            temp = FunToday[activityNum]
            # add the optimal activity choice by finding maximum enjoyment
            temp += max( FunTillToday[(activityNum+1)%3],
FunTillToday[(activityNum+2)%3] )
            maxFun.append(temp)

        # return the values of maximum enjoyment values
        return maxFun
```

EXPLANATION:

We Keep track of the max possible enjoyment possible for a given choice of activity on a given day. Keep updating this with the optimal choices for next day and by the end of day N, we will have a list of the maximum enjoyment that each choice CAN produce.

FOR AN ACTIVITY ON DAY X:

To calculate the enjoyment score, we take the max enjoyment from DAY X-1.

Then we add the maximum value from today's activities. We only consider activities that are not the current activity.

eg:

Max enjoyment for activity 1 on day x

=

activity 1 day X enjoyment value

+

Maximum {

Activity 2 MAX score on day X-1,

Activity 3 MAX score on day X-1,

}

Activities	I	II	III	
	0	0	0	MAX FUN TILL DAY 1
Day 1	1	2	1	
	1	2	1	MAX FUN TILL DAY 2
Day 2	3	5	2	
	5	6	4	MAX FUN TILL DAY 3
	55	32	12	MAX FUN TILL DAY N
Day N	3	5	2	
MAX FUN	35	60	57	