import pandas as pd

import math

from datetime import datetime

def solution(fees, records):

answer = []

records\_array = [record.split(" ") for record in records]

df = pd.DataFrame(records\_array, columns=[

"time", "plate\_number", "in\_out"])

df['time'] = pd.to\_datetime(df['time'], format="%H:%M")

grouped = df.groupby("plate\_number").apply(

fillna).reset\_index(drop=True)

durations = grouped.groupby("plate\_number").apply(get\_duration)

answer = [charging\_fees(

x, fees[0], fees[1], fees[2], fees[3]) for x in durations]

return answer

def fillna(group):

if len(group['in\_out']) % 2 == 1:

record = {"in\_out": "OUT", "plate\_number": group['plate\_number'].iloc[0], "time": pd.Timestamp(

"1900-01-01 23:59:00")}

group = pd.concat(

[group, pd.DataFrame([record])], ignore\_index=True)

return group

def charging\_fees(used\_min, base\_min, base\_price, unit\_min, price\_per\_unit):

if used\_min <= base\_min:

return base\_price

return base\_price + math.ceil((used\_min - base\_min) / unit\_min ) \* price\_per\_unit

def get\_duration(group):

ins = group[group['in\_out'] == 'IN']

outs = group[group['in\_out'] == 'OUT']

duration = (outs['time'].reset\_index(drop=True) -

ins['time'].reset\_index(drop=True)).sum().total\_seconds()/60

return int(duration)