Determine the peak memory used by each group/user inside Google ~~(the~~

~~motivation is determining borg memory resources needed per mdb user)~~.

You are given a single log file with records with the following records:

struct Job {

string job; // candidate should realize this isn't important

string user;

double ram;

double cpu; // don't care about this, skip if desired

double disk; // don't care about this, skip if desired

double start;

double end;

}

I say the file has a month's worth of data for all jobs at Google, where

N = # of records = 10M

U = # of users = 10K

Assume each jobs uses a constant amount of RAM the whole time.

If Job J1 uses R1 of RAM and J2 uses R2 of RAM, if they overlap then

R1 + R2 RAM is needed.

input = [

{“job”: “a1”, “user”: “modou”, “ram”: 50, “start”: 10000001, “end”: 10000002},

{“job”: “a2”, “user”: “modou”, “ram”: 50, “start”: 10000000, “end”: 10000008}

{“job”: “a3”, “user”: “jeff”, “ram”: 50, “start”: 10000000, “end”: 10000008}

]

def peak\_memory\_per\_user(input):

return //

output = {

“modou”: 100,

“jeff”: 50,

}

assert peak\_memory\_per\_user(input) == output

// Brute force approach

// Java

// Assume i have all import statements

Import java.util.\*

Import java.io.\*

#define max (a, b)(a > b ?a:b)

Hashmap<String , Integer>() peakUsages = new Hashmap<String, Integer>();

for(obj in input){

}