Hiring Janitors

Dean Capellari, 2019

Introduction

A museum has open job spots for janitors and is currently hiring. They would like to know how many janitors they need to hire and would like a work schedule for the janitors. After the museum reaches out to mathematicians asking for help, you accept and get to work. The museum presents you with this problem:

Problem Statement

Everyday, x guests come to the museum, where $1000 \leq x \leq 1500$. Guests produce $\frac{99x}{100}$ total trash per day. One janitor can clean 100 trash in 1 hour; the maximum work day is four hours per day, and the maximum work week is six days. The max number of janitors is 25, and the minimum number is 1. Find an algorithm using the number of people and the work hours to find the proper number of janitors required to clean the trash.

Results

The original equation I devised for the number of trash:

$$j = \left(\frac{\left(\frac{99x}{100}\right)}{th}\right)$$

List of variables:

- h=hours a day
- j=#of janitors.
- x=#of guests.
- a=hrs on.
- y=total janitors.
- l is janitors off.
- t=#of trash cleanable in 1 hour(important for fatigue),t=100(for now).

For the basic problems, it works. If we set x=1250 and h=4, we get 1237.5/100 which rounded up to the nearest integer is j=4. If y=11, which means we hired 11 janitors, and j=4, we get this table:

Table:

y=11

	Work(j)	off(l)
s	1 2 3 4	5 6 7 8 9 10 11
m	5 6 7 8	9 10 11 1 2 3 4
t	9 10 11 1	2 3 4 5 6 7 8
w	2 3 4 5	6 7 8 9 10 11 1
t	6 7 8 9	10 11 1 2 3 4 5
f	10 11 1 2	3 4 5 6 7 8 9
s	3 4 5 6	7 8 9 10 11 1 2

The general algorithm for the table is you hire at minimum twice the number of janitors, put j# of janitors on duty, and then for the next day take j janitors on duty and the janitors on duty the previous day get the day off. In the table, you have janitors 1, 2, 3, and 4 working, and 5 thru 11 resting. For day two, you have 5, 6, 7, and 8 on the job, and 9, 10, 11, 1, 2, 3, and 4 resting.

Ad campaign

The museum has an ad campaign that brings in z more people per week.

My first equation:

$$x_n = z + x_{(n-1)}$$

After that, you just plug x into

$$j = \left(\frac{\left(\frac{99xn}{100}\right)}{th}\right)$$

And then use the table algorithm to finish it off.

Fatigue

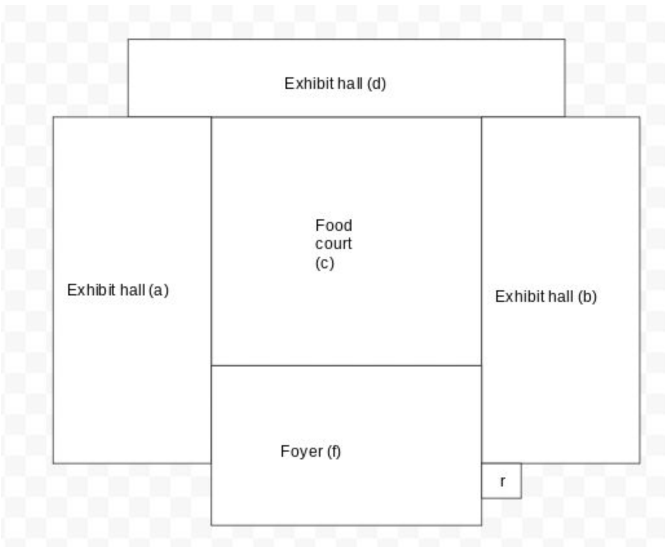
Janitors have now magically leveled up and can now fatigue. Fatigue equation:

$$t = 100 - 10a$$

It's pretty obvious what happens next, although you may need to add a janitor or two depending on the circumstances.

Path Algorithm

This is where you start losing a working, repeatable algorithm. The museum wants you to find a path between rooms so that all trash is cleaned.



Be more flexible when devising paths.
(no legit algorithm)

The trash values for all the rooms are stated below:

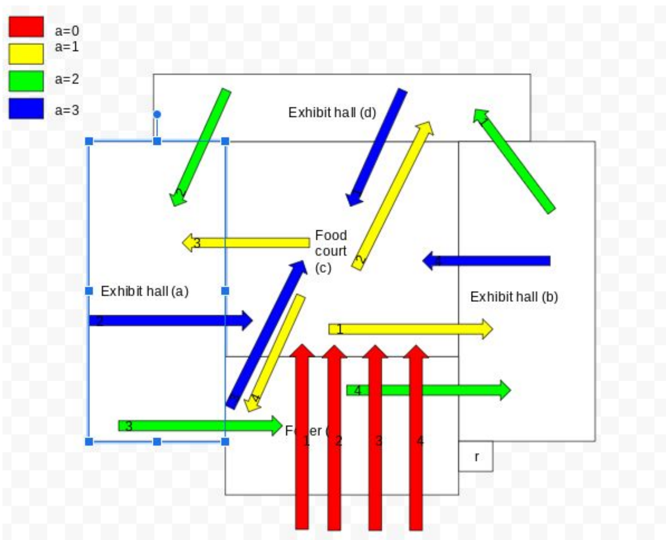
c= $\left(\frac{\left(\frac{99x}{100}\right)}{2}\right)$

a,b,d,f= $\left(\frac{\left(\frac{99x}{100}\right)}{8}\right)$

breakroom(r)=0

This basic process works best for j=4:

a=0: Send everyone to c.
a=1: one to a, one to b, one to d, one to f.
a=2:one to a, one to b, one to d, one to f.
a=3: all to c.
When done go to r.



This process is sort of applicable to all values for j where $1 \le j \le 25$.

1. Send all janitors to dirtiest room.
2. Send as close to an equal number of janitors to each room as you can get.
3. If any trash left in dirtiest room and sending all janitors to dirty room during last hour does not work, send enough janitors to dirty room as needed during step 2.
4. As stated, and if needed, send all janitors to dirtiest room.
5. If finished go to breakroom.

As the diagram shows, it works relatively well for this layout. I have not tried it yet for other j or x values, but it is safe to say it works for this museum.

Conclusion

You can solve it, barely. Maybe if I were to work a bit more on this I would see if it were to be possible to apply this to other jobs in the museum, and even just any business. Regarding constraints, I would probably add in the restriction that janitors can only walk into rooms from adjacent rooms. Other than that, the museum has fully figured out its janitor issue.