

DC Tower Elevator Challenge

Das IBM CIC Büro befindet sich im 35. Stockwerk des DC Towers in Wien. Der DC Tower hat insgesamt 55 Stockwerke. 7 Aufzüge bringen im Dauereinsatz die Angestellten in ihre Büros. Die IBM MitarbeiterInnen erhalten mit Hilfe einer Karte Zutritt in den DC Tower. Beim Durchlass zeigt ein Display, welcher Aufzug sie zu ihrem Büro bringt. Wenn sie das Büro verlassen möchten, drücken sie einen Button und der nächste Aufzug bringt sie ins Erdgeschoss. Die Aufzüge haben keine weiteren Buttons, nur einen Hauptbutton in jedem Stockwerk.

Aufgabe:

Deine Aufgabe ist es, ein Java-Programm für das DC Tower Aufzug-System zu konzipieren und zu implementieren, welches die Aufzuganfragen verwaltet. Eine Anfrage kann sowohl der DC Tower Zutritt im Erdgeschoss, also auch die Rückfahrt vom Stockwerk ins Erdgeschoss sein.

Nachdem das System eine Anfrage bekommen hat, sucht es nach dem nächsten freien Aufzug. Ist keiner sofort frei, soll das System den nächsten einplanen, um die Passagiere abzuholen. Eine Anfrage besteht aus dem aktuellen Stock, dem Zielstock und der Fahrtrichtung als Input.

Input / Output Beispiele:

- DC Tower Zutritt: [current floor: 0, destination floor: 35, direction: UP]
- Rückfahrt vom 14. Stock ins Erdgeschoss: [current floor: 14, destination floor: 0, direction: DOWN]
- Rückfahrt vom 35. Stock ins Erdgeschoss: [current floor: 35, destination floor: 0, direction: DOWN]

Für die Logik wie die Aufzüge sich bewegen können, kannst du aus zwei Varianten wählen:

1. Eine Aufzugfahrt kann nur zwischen zwei Stockwerken verkehren (z. B. 0. -> 35. oder 42. -> 0.).
2. Eine Aufzugfahrt kann auch Zwischenstopps haben (z. B. 0. -> 16. -> 24. -> 35. oder 48. -> 15. -> 0.).

Tipps und Hilfe:

- Du kannst uns den Code gezippt per Email, über Github oder ein anderes Repository schicken.
- Schreibe den Code objektorientiert.
- Wenn eine Anfrage ins System kommt, gilt es für das gesamte System und nicht nur für einen Aufzug. Das liegt daran, dass die Aufzüge keine eigenen Anfrage-Buttons haben.
- Mache dir keine Gedanken über die Türen der Aufzüge.
- Mache dir auch keine Gedanken über die Edge-Cases. Für uns ist es wichtig zu erkennen, dass du die Grundlagen von Java beherrschst.
- Du kannst den Stand des Aufzugs mittels verschiedener Strategien simulieren. Eine wäre mittels Java Thread, aber wenn du dich damit nicht auskennst, kannst du einfach bei jeder Anfrage den Stand manuell aktualisieren.
- Wie oben schon erwähnt, wenn du dich mit Java Threads nicht auskennst, mache dir keine Sorgen. Du musst nicht die Bewegung der Aufzüge simulieren. Es wäre ein Bonuspunkt, wenn du das implementierst.
- Das Testing sollte ungefähr so aussehen:

```
{  
    addRequest(...);  
    addRequest(...);  
    checkAvailableElevators();  
    addRequest(...);  
    ...  
}
```

Wenn du die neuen Anfragen mit einem separaten Thread prüfst, dann ungefähr so:

```
{  
    addRequest(...);  
    addRequest(...);  
    ...  
}
```

Wenn du etwas nicht verstehst, schreib einfach auf, wie und wieso du was implementiert hast. Wir akzeptieren jede Lösung.