Catalysts

software is our passion

Catalysts Coding Contest

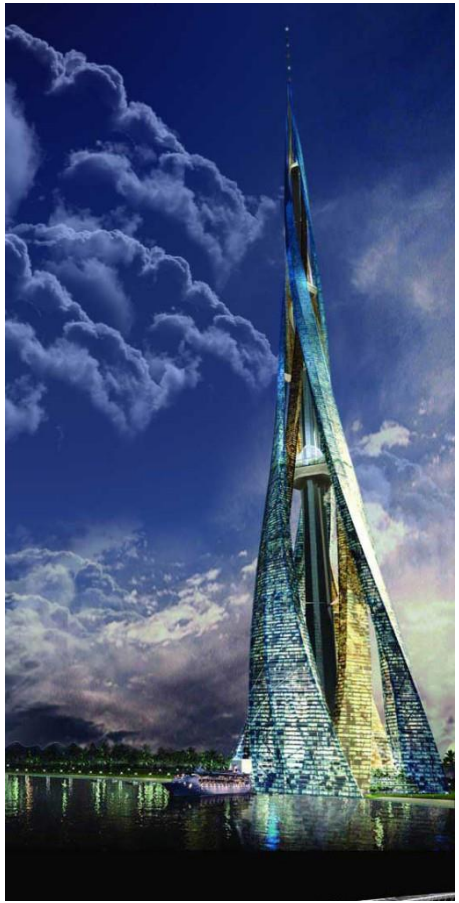
Vienna / Austria and Cluj / Romania

October 11th, 2013

Elevator control system for a 2-km skyscraper

Jeder kennt sie. Jeder verwendet sie mehr oder weniger oft. Doch wirklich unerlässlich werden sie mit steigender Gebäudehöhe: *Aufzüge*

Und bei den stetig wachsenden Riesen-Wolkenkratzern der letzten und kommenden Dekaden, wie dem als 2 km „Ungetüm“ geplanten „Dubai City Tower“, werden die Aufzugssysteme und deren Steuerung eine immer herausforderndere Aufgabe.

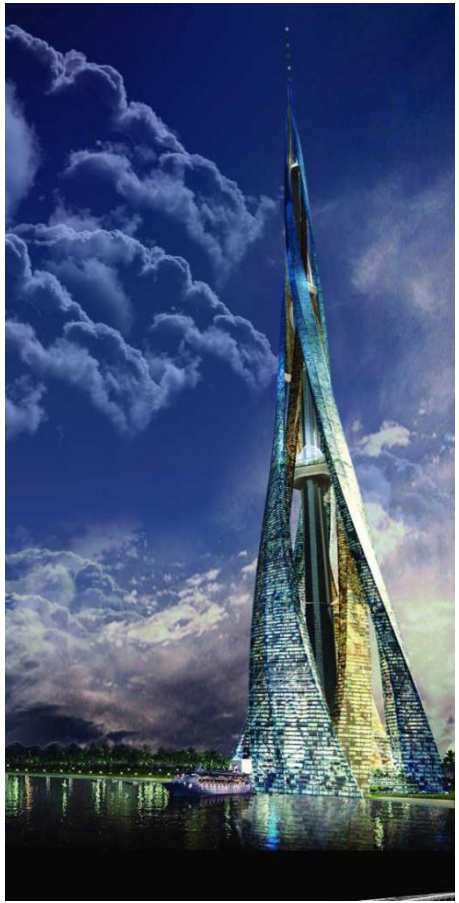


Elevator control system for a 2 km high skyscraper

Everyone knows them. Everyone uses them more or less frequently. But everytime a building is higher than another one, they get more and more crucial: *Elevators*

And indeed: there seems to be no end when engineers around the world plan such new mega-skyscrapers like the planned, 2 km high „Dubai City Tower“.

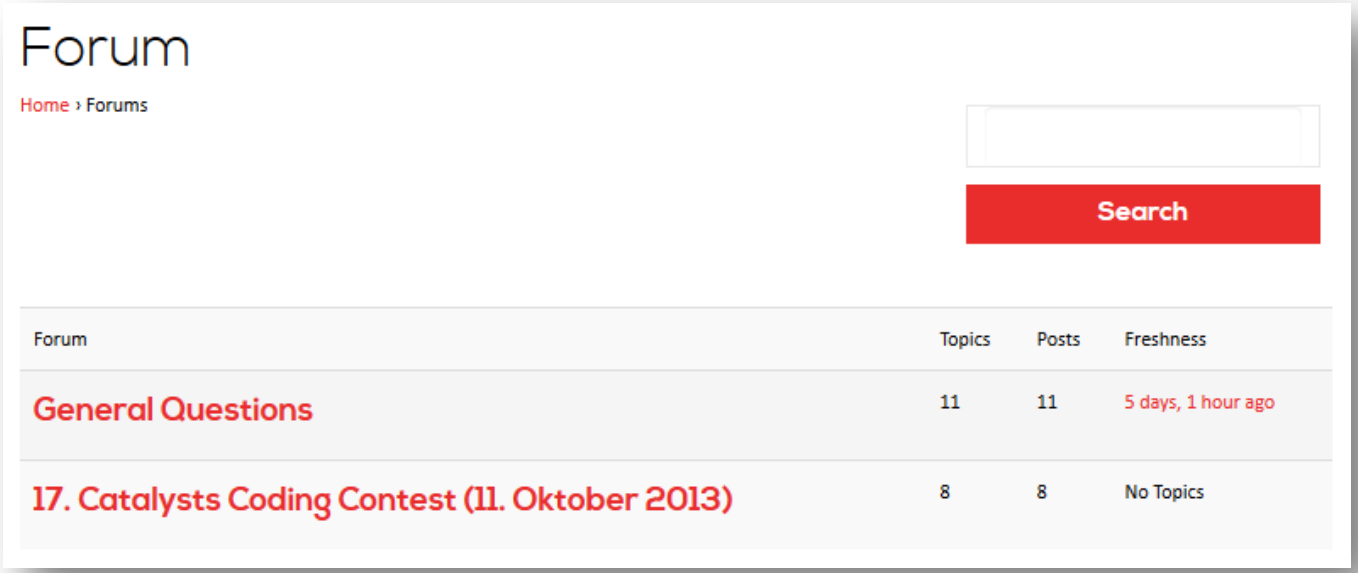
And to control the elevator systems for such a huge „monster“ can become really exciting.



Question & Answer Forum

Deutsch http://contest.catalysts.cc/forums/

English http://contest.catalysts.cc/en/forums/



Elevator control system for a 2-km skyscraper

Input-, Output-Einheiten

- Zeitstempel sind in **Millisekunden** [64bit integer]
- Geschwindigkeit ist in m/**s** [floating point]
- Beschleunigung & Abbremswirkung ist in m/**s²** [floating point]

Vereinfachung

- Lift-Ein- und Ausstiegs-Dauer wird ignoriert

Globale Konstanten

- Wolkenkratzer hat 400 Stockwerke
- Ein Stockwerk ist 4.5m hoch
- Aufzüge starten im niedrigsten erreichbaren Stockwerk

Input-, output-units

- Timestamps are in **milliseconds** [64bit integer]
- velocity is in m/**s** [floating point]
- acceleration and deceleration are in m/**s²** [floating point]

Simplification

- Lift entry and exit duration is ignored

Global constants

- Skyscraper has 400 floors
- A floor is 4.5m high
- Elevators start in the lowest reachable floor

Level 1 – Travel time

The target of this level is to calculate the travel times of given distances.

Use floating point arithmetic during the calculation and round up at the end to a 64 bit integer. (1000.2 → 1001; 1000.6 → 1001)

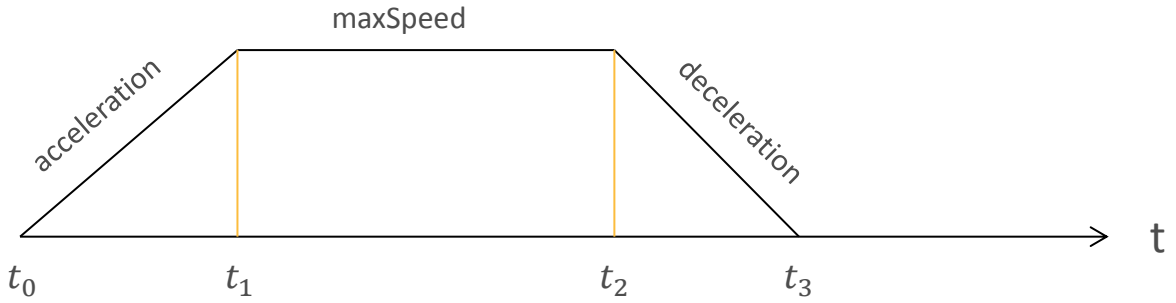
There are two possible situations which must be handled differently

1. Distance is higher or equal than maxAccelerationDistance + maxDecelerationDistance
2. Distance is lower than maxAccelerationDistance + maxDecelerationDistance

$$maxAccelerationDistance = \frac{acceleration}{2} * \left(\frac{maxSpeed}{acceleration}\right)^2$$

$$maxDecelerationDistance = \frac{deceleration}{2} * \left(\frac{maxSpeed}{deceleration}\right)^2$$

Case 1 – Higher or equal



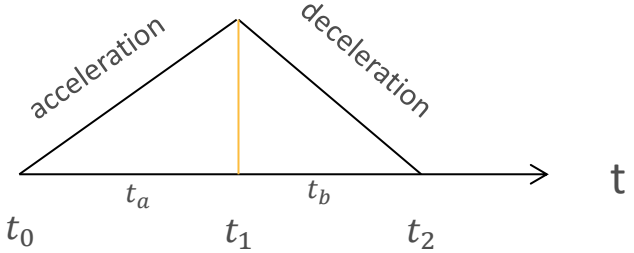
$$t_0 = 0$$

$$t_1 = t_0 + maxAccelerationTime$$

$$t_2 = t_1 + \frac{maxSpeedDistance}{maxSpeed}$$

$$t_3 = t_2 + maxDecelerationTime$$

Case 2 – lower



$t_0 = 0$

$t_1 = t_0 + accelerationTime$

$t_2 = t_1 + decelerationTime$

Additional formulas

$$t_a = \sqrt{\frac{distance}{\frac{acceleration}{2} + \frac{deceleration}{2} * \left(\frac{acceleration}{deceleration}\right)^2}}$$

$$t_b = \frac{acceleration * t_a}{deceleration}$$

Input & Output

Input

acceleration maxSpeed deceleration distance

Output

travelTime

Example input

2.1 10.0 2.9 9.0

Example output

3845

Note

A difference of 2 milliseconds is tolerated