

2021

# מטלה 0 (חלק 1):

תכנות מונחה עצמים

מגישים : דניאל זקן 207296989 , דולב דובלון 207867342

## סקירה ספרותית

קישור 1 : <https://www.youtube.com/watch?v=sigiJAJWUVg>

סרטון מקיף ב YouTube שעוסק בתכנון מערכת מעליות.

מחבר הסרטון : think software

מהסרטון ניתן ללמוד הרבה על הבעיה , ניתן להבין שישנם סוגים שונים של מעליות, סוגים שונים של אלגוריתמים והיבטים רבים לחשוב ולתכנן מערכת מעליות.

קישור 2 :

<https://medium.com/geekculture/system-design-elevator-system-design-interview-question-6e8d03ce1b44>

מאמר מקיף שעוסק במידול הבעיה. נותן דוגמא לדיאגרמה UML ונותן היבט נוסף לבעיה ברמת שלבי הפעולה של המעלית , הגעה פתיחת דלתות יציאה לתנועה, מצבים שקורים בבעיה.

קישור 3 :

<https://github.com/Cleod9/elevatorjs>

<https://github.com/tomtel14/elevator-simulation>

מהסימולציה ומהקוד ניתן לשחק וכך להבין מה בדיוק ההיבטים שיש להתייחס למעליות ותנאי הסביבה שמתקיימים בעולם הבעיה.

קישור לעצמנו למחוק בסוף

<https://www.hellocodeclub.com/design-elevator-system-java/#Conclusion>

# חלק 1: מידול בעיה, תכנון, דיאגרמת מחלקות.

## מידול והבנת הבעיה:

### דרישות המטלה:

לכל בניין יש קומת מינימום ומקסימום וידוע כמות המעליות

לכל מעלית יש מאפיינים של זמן עצירה, זמן תחילת תנועה, ומהירות (כמה שניות לוקח לה לעבור לקומה).

המשתמש צריך להקיש מחוץ למעליות את קומת היעד (destination) ואז המערכת צריכה לשבץ (להקצות) לו מעלית מסוימת אשר מוגדרת לה ולעצור בקומת היעד.

המערכת תרצה לשבץ את המעלית שתצמצם למינימום את זמן ההגעה (זמן ההגעה מוגדר להיות משך הזמן בשניות שבין הקריאה למעלית ובין ההגעה לקומת היעד)

בהינתן אוסף של קריאות למעליות בזמן נרצה להגדיר אסטרטגיית שיבוץ מעליות לקריאות שתצמצם למינימום את **סך משך זמן ההגעה עבור כלל הקריאות**.

קומות מגורים זה 1 ומעלה

קומות כניסה או יציאה לבניין זה 0 ומטה

ואין הגבלת כמות אנשים לכל מעלית.

בפתרון המלא (אלג' online) ברגע שמתקבלת קריאה (זימון) יש לקחת בחשבון באופן מידי את הקריאה, ובהתאם לקבל החלטה לפי האלגוריתם.

ניתן לעצור את המעלית בקומת ביניים, לדוגמא, אם המעלית עולה מקומה 1 לקומה 12 ניתן לעצור אותה בקומה 5 אם היא נמצאת בקומה קטנה או שווה ל 5, אבל לא אחרי זה.

כל מעלית יכולה להיות במצב "עלייה", "ירידה" או "מנוחה". ויש את המאפיינים הבאים, זמן עצירה, זמן תחילת תנועה, ומהירות.

לנסח אלגוריתם offline . אלגוריתם online

## מידול

דברים שצריך לקחת בחשבון:

\*מעלית יכולה לאסוף נוסעים רק כאשר היא עוצרת בקומה.

\*מספר הקומות בבניין , אל מול מספר המעליות.

\*אנחנו רוצים לקבל את סך משך זמן ההגעה עבור כלל הקריאות האופטימלי.

ננסה לחשוב על הבעיה מכמה נקודות מבט, אם יש לבניין מספר רב של קומות למשל אז 100 יש אפשרות לחלק את המעליות ל 4 אזורי אחריות למשל כל 25 קומות הן אזור. ובכך ניתן לתת מענה מהיר יותר לקריאות. בנוסף ניתן להשתמש ביוריסטיקות פעולה עבור זמני פעילות המעליות כאשר צריך לתעדף מעליות למקומות שונים לפי הזמן ביום, למשל בבוקר יש לשמור יותר מעליות בקומות הכניסה , ובערב יש לפזר את המעליות באופן אחיד בקומות העליונות על מנת לאפשר יציאה מהירה מהבניין.

מרכיבי הבעיה :

1. נוסע

2. מעלית

3. לכל מעלית יש דלת המעלית

4. כפתורי ניווט בכניסה למעלית

5. מחשב שמנהל את מערכת המעליות

6. קומות

7. ועוד...

קריאה למעלית :

בעת קריאה למעלית נדרש לבצע חישוב המסלול האופטימלי וכך נגיע לתוצאה הרצויה.

## אלגוריתם offline

המטרה : לתכנן את המסלול כך **סך משך זמן ההגעה עבור כלל הקריאות** יהיה האופטימלי ביותר.

נתונים : כמות המעליות ידועה ומיקומן ומצבן.

### שלב א :

1. לוקחים את כל הקריאות ואת כל המעליות.
2. מחשבים מה המרחק הכי קצר שיכול להיות בין קריאה למעלית רק עבור מעליות שלא מתואמות עם קריאות אחרות. אם יש שני אנשים באותה קומה נחלק למקרים (\*).
3. קיבלנו את המהירות הכי קצרה שמעלית יכולה להגיע לקומה של הבן אדם שקורא לה, נתאים אותה לקריאה.
4. אם קיבלנו שניים עם אותו מרחק בין מעלית לקריאה, נחלק למקרים :
  - (a) אם שניהם עולים אז נתאים את המעלית לקריאה התחתונה.
  - (b) אם שניהם יורדים אז נתאים את המעלית לקריאה העליונה.
  - (c) אם היעד של שניהם הוא לכיוונים שונים, ניקח את הקריאה שבכיוון שלה, יש פחות מעליות כדי שהיא "תאסוף על הדרך" יותר אנשים.
5. אם יש עוד מעליות ולא נעצרנו בשלב 2 אז נחזור על פעולות 1 – 4 שוב כאשר נוריד את הקריאה שכבר הצמדנו לה מעלית.

### שלב ב :

- אם הגענו לשלב ב' זה אומר שלא משנה אם יש מעלית פנויה שקושרה אליה כבר קריאה, או שאין מעלית פנויה כי אם היה יעיל לקרוא לה לקריאה, אז כבר היינו עושים את זה בשלב א'.
- כעת אנחנו נרצה להשתמש במעליות שכבר קישרנו להן קריאה.
1. שוב נחשב את זמן המסלול בין כל המעליות (רק המעליות שכבר הותאמה להן קריאה) לכל הקריאות שעדיין לא טופלו, אחרי שהוספנו למסלול את זמן ההגעה לקריאה המקורית של המעלית, חישוב הזמן מתבצע מהקומה של הקריאה המקורית עד לקריאה הנוכחית, (המרחק בין המעלית לקריאה הראשונה + המרחק בין הקריאה הראשונה לקריאה שאחריה).
  2. ניקח את המרחק הכי קצר ונוסיף את הקריאה של המרחק הכי קצר למסלול של המעלית שמותאמת לה **רק אם היעד של הקריאה הוא לאותו כיוון** וגם הוא לא גבוהה יותר או נמוך בהתאמה לכיוון היעד של הקריאה הסופית לפני ההוספה, נמחק קריאה זו מרשימת הקריאות וכעת נחזור לשלב א'. גם פה נתחשב במקרה של שני אנשים ומעלה באותה קומה (\*)
  3. אם לא נשאר יעדים באותו כיוון שמתאימים, אז נשלח את המעליות לעשות את המסלולים שכבר קבענו להן, ולאחר שיגיעו ליעד של סוף המסלול באותו הכיוון, נחזור לשלב א'.

(\*) חלוקה למקרים כאשר שני אנשים ומעלה באותה קומה:

- (a) אם כל האנשים עולים אז הם יעלו על אותה מעלית.
- (b) אם כל האנשים יורדים , אז הם יעלו על אותה מעלית.
- (c) אם הכיוונים של היעדים מנוגדים : נתאים מעלית שכיוון המסלול שלה הוא אותו כיוון.

## אלגוריתם online

האלגוריתם מתחיל בכך שמתבצעת האזנה לקריאות בזמן אמת.

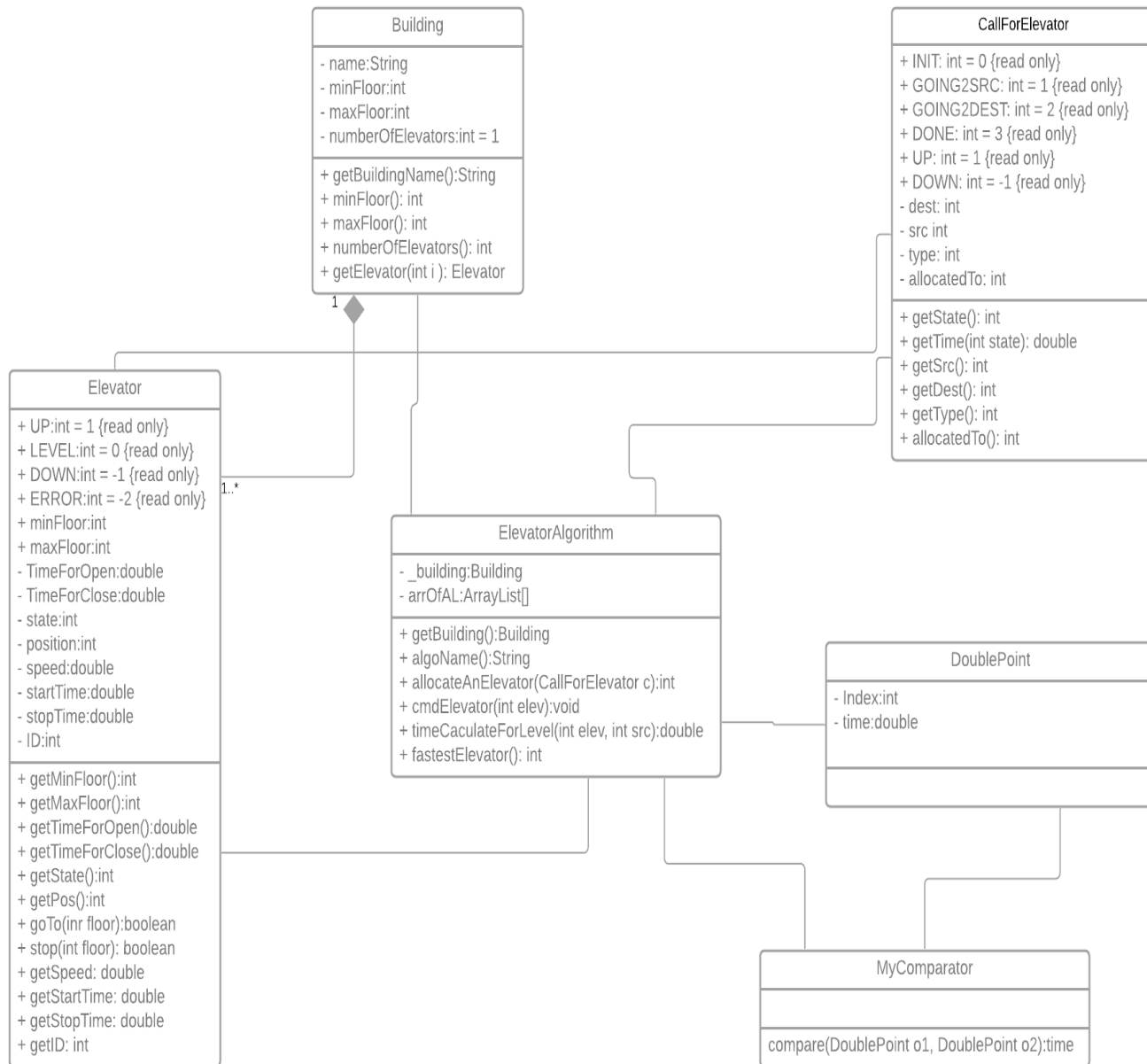
אלגוריתם און ליין שונה בכך שהנתונים לא זמינים מההתחלה אלה מגיעים לאורך זמן, לכן השינוי שנוסיף לאלגוריתם offline על מנת שיוכל להתמודד עם מידע שמגיע בהקצבה , הוא שאם המעלית בתנועה ומקבלת קריאה נחשב את המעלית הכי קרובה אליה מבין המעליות שבכיוון שלה , ובין המעליות שבמנוחה. אחרת נחשב את המסלול הטוב ביותר מבין המעליות שלא בכיוון שלה.

בפועל:

עבור כל קריאה ניקח את כל המעליות ונחשב מי המעלית שמגיעה הכי מהר וגם בכיוון נסיעה שלה(אם לא במנוחה)וניקח את הכי מהירה שמקיימת את התנאים(אם הכי מהירה לא בכיוון נסיעה אז נבדוק את הכי מהירה השנייה וכך הלאה)אם אין ניקח את המעלית הכי מהירה כברירת מחדל.

בחישוב הזמן נתייחס לזמן פתיחה,סגירה, עצירה והתחלה של המעלית וכמובן חישוב של הזמן בהתבסס המרחק והמהירות של כל מעלית שהיא משתנה בין מעלית למעלית.

## UML Diagram



-- link to the diagram.

[https://lucid.app/lucidchart/367d4a61-87a4-4e4c-9767-a1dd49466575/edit?view\\_items=fqu mZbE4oneC&invitationId=inv\\_5034536b-bfcf-4c2a-8c27-61bfd397e94f](https://lucid.app/lucidchart/367d4a61-87a4-4e4c-9767-a1dd49466575/edit?view_items=fqu mZbE4oneC&invitationId=inv_5034536b-bfcf-4c2a-8c27-61bfd397e94f)

## סטטים עבור - offline

תזכורת...

לכל מעלית יש מאפיינים של זמן עצירה, זמן תחילת תנועה, ומהירות (כמה שניות לוקח לה לעבור לקומה).

סטט 1

קומה	מפלס	מפלס	מפלס	מפלס
10				
9				
8				
7				
6				
5				
4				
3				
2				
1				
0	א	א	א	א
-1				
-2				

בניין 10 קומות + קומת קרקע שתי קומות חניון.

מעלית אחת בקומה 0 בן אדם אחד קומה 0 צריך להגיע לקומה 5

1. הבן אדם עולה על המעלית

2. המעלית עולה לקומה 5

3. המעלית עוצרת

הבדיקה תהיה האם הזמן הכולל שלקח היה, זמן תחילת תנועה + מעבר 5 קומות + זמן עצירה

בנוסף הקריאה תצורף למעלית למעלית a.

## סטט 2

קומה	מפלס	מפלס	מפלס	מפלס
10				
9				
8				
7				
6				
5				
4				
3				
2				
1				
0	א	א	א	א
-1				
-2				

בניין 10 קומות + קומת קרקע שתי קומות חניון.

מעלית a בקומה 2 בן אדם אחד קומה 0 צריך להגיע לקומה 5

1. המעלית יורדת לקומה 0

2. הבן אדם עולה על המעלית

3. המעלית עולה לקומה 5

4. המעלית עוצרת

הבדיקה תהיה האם הזמן הכולל שלקח היה, זמן תחילת תנועה + מעבר 2 קומות + זמן עצירה + זמן תחילת תנועה + מעבר 5 קומות + זמן עצירה.

והקריאה תצורף למעלית a.



### טסט 3

בניין 10 קומות + קומת קרקע שתי קומות חניון.

מעלית a בקומה 2 בן אדם אחד קומה 0 צריך להגיע לקומה 5 בן אדם שני בקומה 5 צריך להגיע לקומה 8

1. המעלית יורדת לקומה 0
2. הבן אדם עולה על המעלית
3. המעלית עולה לקומה 4
4. המעלית עוצרת
5. אוספת את בן אדם בקומה 4
6. המעלית עולה לקומה 5
7. המעלית עוצרת עוצרת בקומה 5
8. המעלית עולה לקומה 8
9. המעלית עוצרת בקומה 8

הבדיקה תהיה האם הזמן הכולל שלקח היה, זמן תחילת תנועה + מעבר 2 קומות + זמן עצירה + זמן תחילת תנועה + מעבר 4 קומות + זמן עצירה + זמן תחילת תנועה + מעבר 3 קומה + 1 זמן עצירה + זמן תחילת תנועה + זמן מעבר 3 קומות + זמן עצירה.

הקריאה מקומה 0 תצורף למעלית a ולאחר מכן הקריאה מקומה 4 תצורף למעלית a

### טסט 4

בניין 10 קומות + קומת קרקע שתי קומות חניון.

מעלית a בקומה 2 מעלית b בקומה 1 בן אדם אחד קומה 0 צריך להגיע לקומה 5 בן אדם שני בקומה 5 צריך להגיע לקומה 8

1. מעלית b יורדת לקומה 0
2. בן אדם עולה על מעלית
3. מעלית b עולה לקומה 5
4. מעלית a עולה לקומה 4
5. אוספת את הבן אדם ועולה לקומה 8

מעלית a ומעלית b מבצעות אותן כמות פעולות

עצירה : 2 ; תחילת תנועה : 2 ; מעבר קומה : 6

ולכן הבדיקה תהיה האם זמן המערכת הוא:

$2 * \text{זמן עצירה} + 2 * \text{זמן תחילת תנועה} + 6 * \text{זמן מעבר קומה}$ .

בנוסף הקריאה מקומה 2 תצורף למעלית b והקריאה מקומה 4 תצורף למעלית a

קומה	מעלית	מעלית	מעלית	קומה
10				10
9				9
8				8
7				7
6				6
5				5
4				4
3				3
2		a		2
1				1
0				0
-1				-1
-2				-2

קומה	מעלית	מעלית	מעלית	קומה
10				10
9				9
8				8
7				7
6				6
5				5
4				4
3				3
2		a		2
1	b			1
0				0
-1				-1
-2				-2

## טסט 5

בניין 10 קומות + קומת קרקע שתי קומות חניון.

מעלית a בקומה 2 מעלית b בקומה 1 בן אדם אחד קומה 0 צריך להגיע לקומה 5 בן אדם שני בקומה 5 צריך להגיע לקומה 8 בן אדם שלישי בקומה 10 צריך להגיע לקומה 0

מעלית a מבצעת את התנועה הכי ארוכה ולכן נתייחס אליה כאל סיום זמן סך זמן הקריאות ולכן הזמן יהיה

$4 * \text{זמן עצירה} + 4 * \text{זמן תחילת תנועה} + 18 * \text{זמן מעבר קומה}$ .

הקריאה בקומה 0 תצורף למעלית b, הקריאה בקומה 4 תצורף למעלית a ולאחר מכן הקריאה בקומה 10 תצורף למעלית a.

קומה	מעלית a	מעלית b	אדם
10			↓
9			
8			
7			
6			
5			
4			↑
3			
	a		
1		b	
0			↑
-1			
-2			