

אתגר המעלית

הסבר כללי

- יש לתכנן ולבנות מערכת מעליות **יעילה** תוך שימוש בשפת python.
- לאחר הרצת הקוד יראה המשתמש את הבניין שיכלול את הקומות, המעליות ופקדי המעליות.
- בכל קומה יוצג פקד קריאה למעלית שעליו כתוב מספר הקומה. לחיצה על הכפתור תזמין מעלית לקומה (**גם אם אין אף מעלית פנויה כרגע**).
- עם הזמנת מעלית לקומה, יוצג ליד כפתור המעלית מספר יורד המייצג את מספר השניות שנותרו עד להגעת המעלית.**
- על אלגוריתם המעליות להביא לזמן ההמתנה המינימלי האפשרי למעלית מבלי להאריך את זמני ההמתנה של אלה שכבר הזמינו מעלית.**
- יש לאפשר הגדרה/שינוי של מספר הקומות ומספר המעליות בבניין באופן פשוט (באמצעות שינוי הגדרות בקוד או קובץ הגדרות).
- המעליות ינועו במהירות של **חצי שניה לקומה** (יש להציג את התנועה באנימציה חלקה – ולא למשל קפיצות בין הקומות), וכאשר יגיעו לקומת היעד (אליה הזמנה המעלית), יתעכבו במשך שתי שניות.
- סרטוני הדגמה נמצאים בתיקייה המצורפת.

הגדרות ממשק משתמש

- כל קומה תוצג עם רקע של לבנים. דוגמא בהמשך מסמך זה.
- בין כל שתי קומות יוצג פס שחור בעובי של **7** פיקסלים שיחושב כחלק מגובה הקומה שמתחתיו.
- המעלית תיוצג ע"י שימוש בתמונה נתונה (קובץ **elv.png**).
- בעת הגעת מעלית לקומה יש להשמיע קול נתון (קובץ **ding.mp3**).
- עם הלחיצה עליו, יש לצבוע את טקסט פקד הקריאה למעלית בירוק ולהחזירו לצבעו הרגיל עם הגעת המעלית אליו.

הערות

- את קובצי העזר ניתן למצוא בתיקיה.
- הפתרון צריך לכלול שימוש ב- best practices הרלוונטיים לפיתוח.
- חובה להשתמש במתודולוגיית **OOP** (כל אייטם חייב להיות אובייקט).
- יש לצרף מסמך קצר באנגלית המסביר על הארכיטקטורה בפרויקט, והסבר על האלגוריתם הראשי שפותח עבור הפרויקט.
- יש להעלות את הפרויקט ל**GitHub** ולהגיש לינק לפרויקט.

דוגמה לפריסת ממשק הבניין והמעליות:

