

תרגול מס' 11 – שימוש ב- UML Statechart דרך כלי CASE

מטרת הדיאגרמה:

מתארת את ההתנהגות הדינמית של אובייקטים ע"י ציון תגובותיהם לאירועים שונים. בדרך כלל משתמשים ב-statechart לייצוג ההתנהגות של classes, אם כי ניתן להשתמש ב-statechart גם לייצוג use-cases, actors, subsystems, או methods.

ניתן לראות Statechart כגרף שמתאר מכונת מצבים:

☒ צמתי הגרף הם המצבים עצמם.

☒ קשתות הגרף הם המעברים האפשריים בין המצבים.

רכיבי הדיאגרמה:

1. מצב (state) הוא תיאור של אובייקט ברגע נתון. מסומן ע"י מלבן עם פינות מעוגלות.

לדוגמא, לטלפון יש שני מצבים: מצלצל ושותק.

☒ פעילות במצב (activity) היא פעולה מתמשכת (ארוכה) הניתנת להפרעה

ע"י אירוע. ניתן להגדיר פעילויות מסוג do (מתבצעות כל זמן שהאובייקט

במצב), entry (מתבצעות עם הכניסה למצב), או exit (מתבצעות עם היציאה מהמצב).

2. מעבר (transition) מגדיר את האירועים והתנאים למעבר ממצב למצב. מעבר

מסומן בחץ, שעליו יכולים להופיע:

event [condition] | action

☒ אירוע (event) הוא דבר שקורה בנקודת זמן מסוימת (משכו זניח ולא ניתן

להפרעה) ויכול לגרום לשינוי במצבו של האובייקט.

סוגי אירועים:

i. אירועים עם שמות – reset, e, call.

ii. אירועים המודיעים על התחלת פעילות – started(A),
stopped(A).

iii. אירועים המודיעים על קיום תנאי – true(c), false(c).

iv. אירועים המודיעים על כניסה למצב – entered(S), exited(S).

v. אירועים המודיעים על פסק זמן – tm(e, n), כאשר e הוא אירוע ו-n הוא הזמן אחריו קורה ה-timeout.

☒ תנאי (condition) הינו ביטוי שניתן להוסיפו למעבר ממצב למצב. התנאי

יבדק כאשר האירוע יתרחש והמעבר יאופשר אם התנאי מתקיים.

הערה: אם מופיע אירוע מפורש על המעבר, התנאי יבדק ברגע האירוע. אם אין אירוע ובמצב (המקור) יש פעילות, התנאי יבדק בתום הפעילות. אחרת (אין אירוע ואין פעילות), התנאי יבדק מייד בכניסה למצב (המקור).

תנאי יכול להיות:

1. פרדיקט – מציין קשרים לוגיים (and, or, not) בין פונקציות בוליאניות.

2. in(S), כאשר S הוא מצב.

3. active(A), כאשר A היא פעילות.

☒ פעולה (action) היא רצף של פקודות שזמן ביצוען זניח ולכן יכולות להתבצע

בזמן מעבר בין מצבים. פעולה הינה יחידה אטומית שאיננה ניתנת להפרעה (מצפים שתתבצע במלואה או לא תתבצע כלל).

פעולה יכולה להיות:

1. יצירת אירוע עם שם generate(reset).

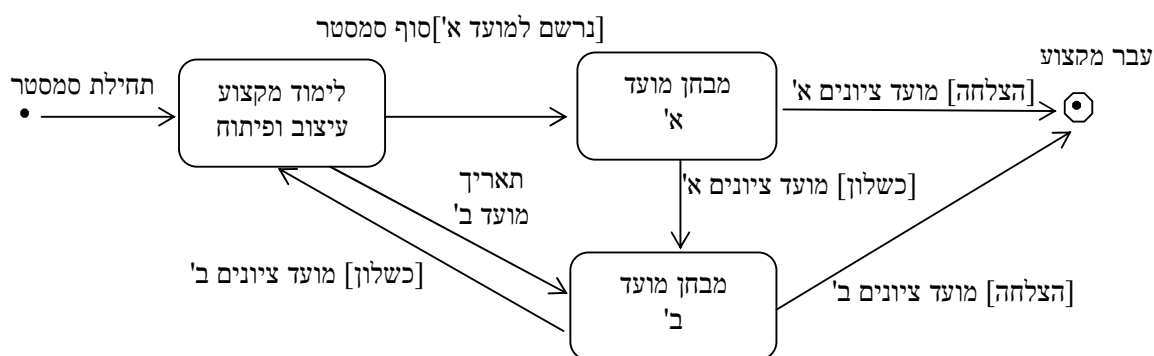
2. השמה למשתנה (x:=x+1).

3. התחלת פעילות (activity) – start(A), stop(A).

4. מחיקת היסטוריה – clh.

5. שרשור פעולות בטור (generate(reset);x:=x+1).

דוגמא 1: תרשים מצבים עבור סטודנט הלומד עיצוב ופיתוח



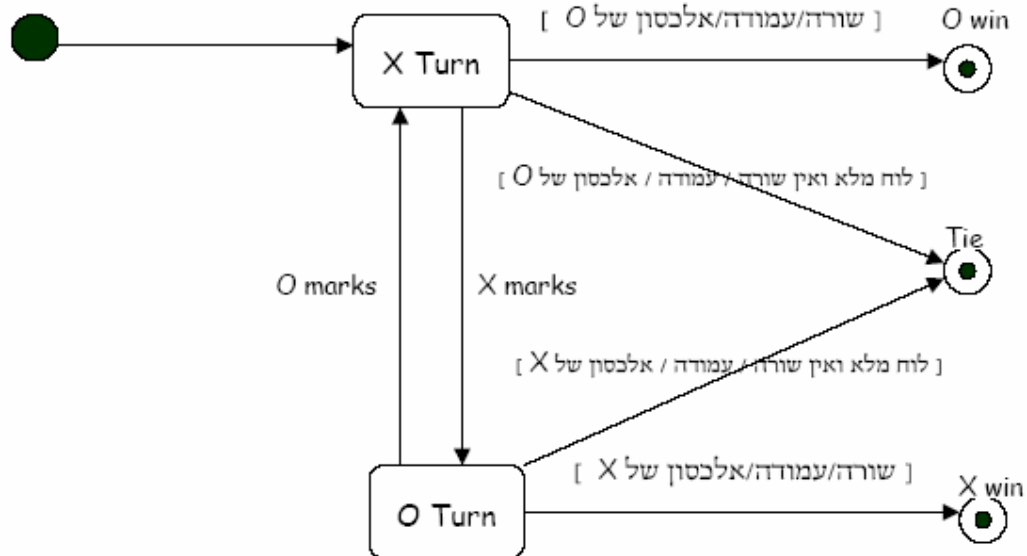
הסברים:

◇ בכל נקודת זמן האובייקט/המערכת נמצאת במצב אחד.

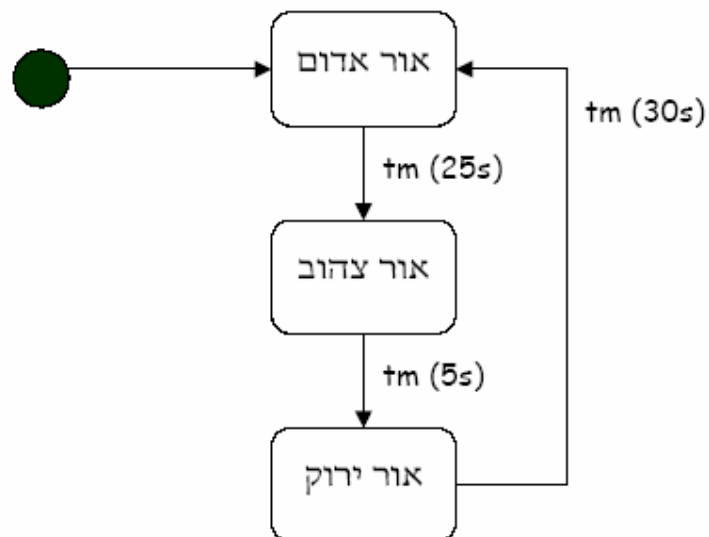
◇ האובייקט/המערכת נשארת במצב עד שאחד המעברים היוצאים מהמצב

מאופשר.

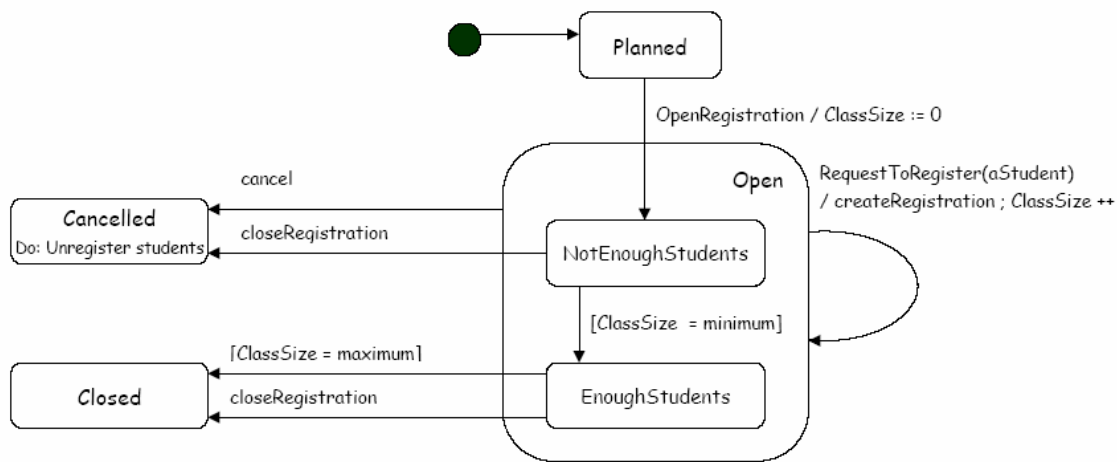
דוגמא 2: תרשים מצבים עבור משחק X-O



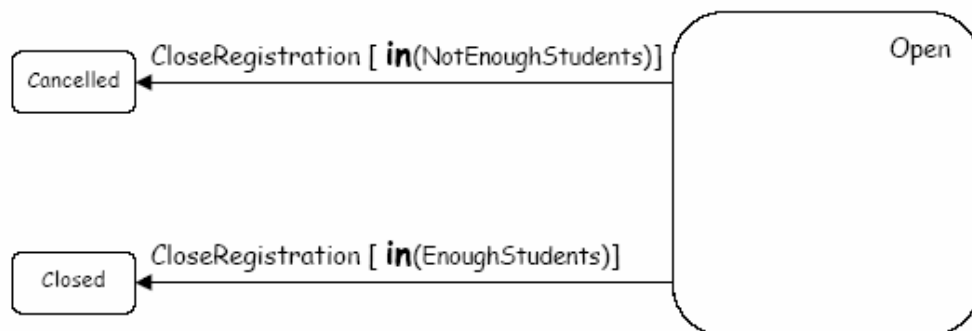
דוגמא 3: תרשים מצבים עבור רמזור



דוגמא 4: תרשים מצבים עבור קורס (כולל רישום לקורס ופתיחתו במידה ונרשמו מספיק סטודנטים).



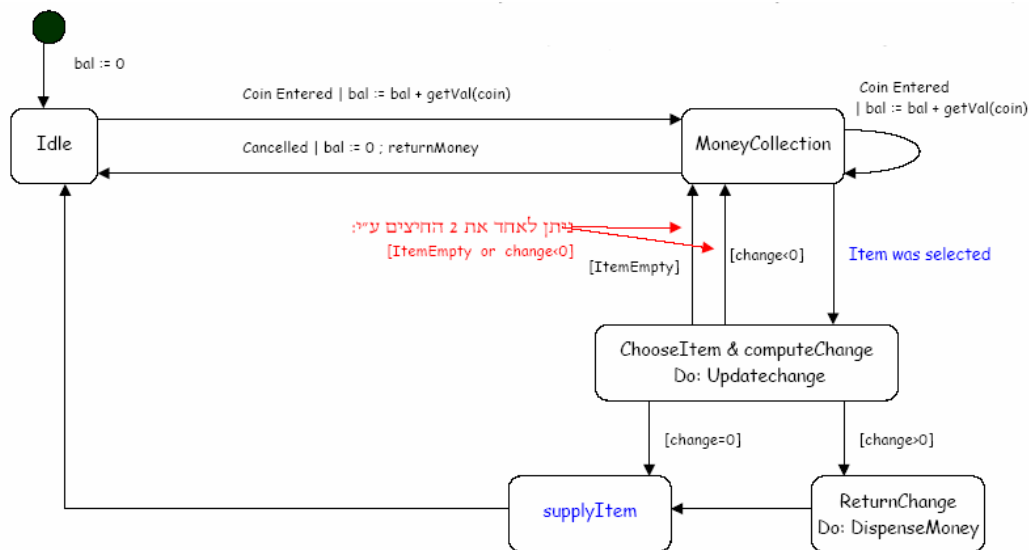
הערה: ניתן למדל את המעבר מ- Open ל- Cancelled ו- Closed גם ע"י שימוש בתנאי מהצורה in(S):



היררכית מצבים - קינון (nesting):

קינון תרשימים מאפשר קריאות טובה יותר. אם נסתכל על אובייקט בעל n תכונות בוליאניות, הרי שיכולים להיות לו 2^n מצבים, דבר שיסבך את קריאות הדיאגרמה. לכן נחלק את הדיאגרמה לרמות שונות, כאשר כל רמה הינה פירוט של הרמה הקודמת (בדומה ל- DFD).

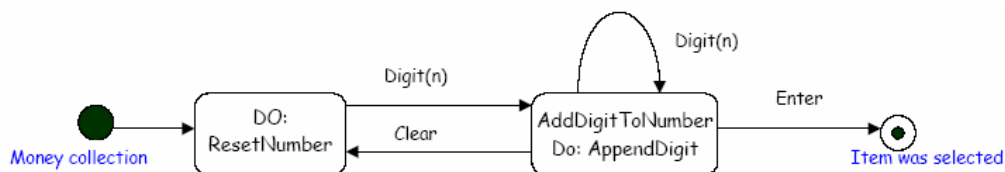
דוגמא 5: תרשים מצבים עבור מכונת משקאות.



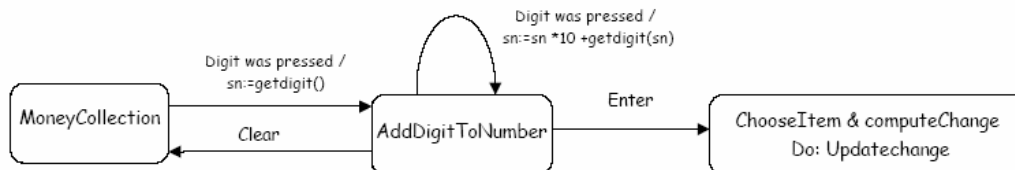
קיטון המעבר: *Item was selected*

האירוע *Item was selected* כולל בתוכו מספר תתי-אירועים:

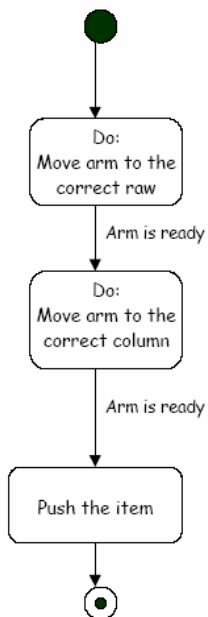
- המשתמש יכול להקיש ספרות.
- המשתמש יכול להתחיל מחדש ע"י לחיצה על **Clear**.
- המשתמש מאשר את המספר (הבחירה הסופית) ע"י לחיצה על **Enter**



אפשרות נוספת:



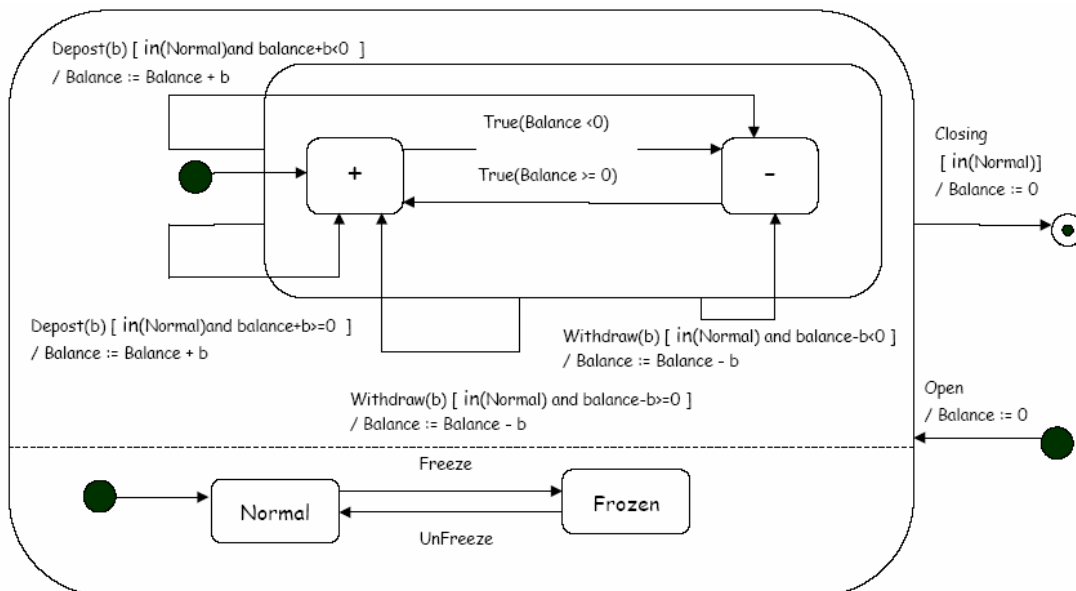
קיטון המצב: [supplyItem](#)




מקביליות:

כדי לתאר מצבים בלתי תלויים של המערכת, נשתמש במקביליות. סימון: קווים מקווקווים בין המצבים הבלתי תלויים.


דוגמא 6: תרשים מצבים עבור חשבון בנק



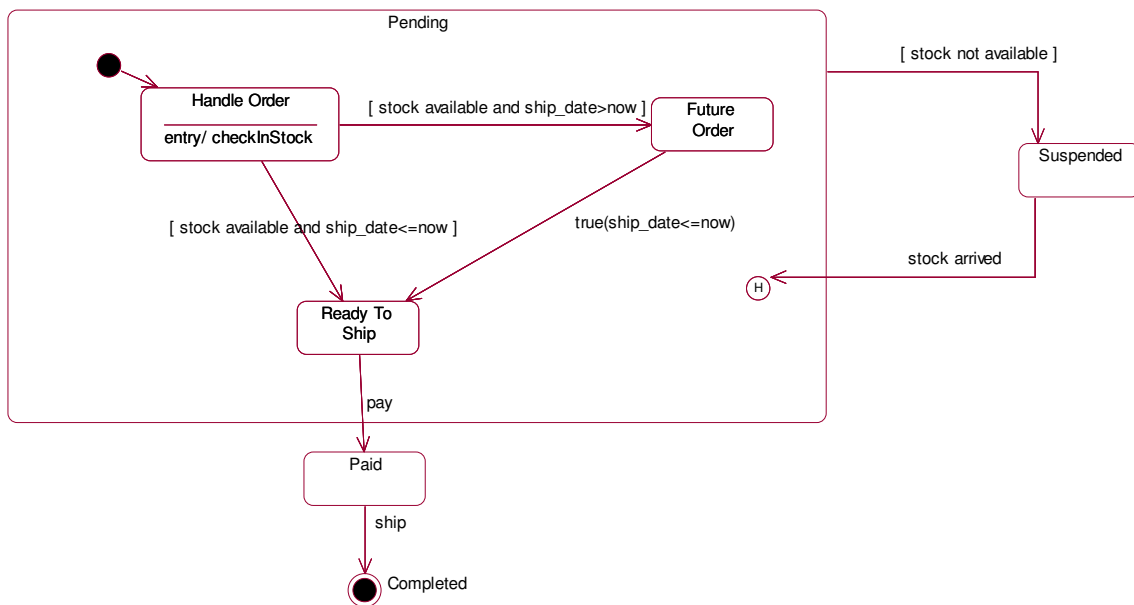
שימוש ב-history:

כאשר מעבר נכנס לתוך מצב מורכב (הכולל תתי-מצבים), הפעולה מתחילה במצב ההתחלתי לפי  (אלא אם המעבר נכנס ישירות למצב כלשהו). ישנם מקרים בהם ברצוננו לתאר מודל של אובייקט המסוגל לזכור את תת-המצב האחרון בתוך מצב מורכב, בו הוא שהה לפני עזיבתו את המצב המורכב. במקרה כזה נשתמש בסימון:

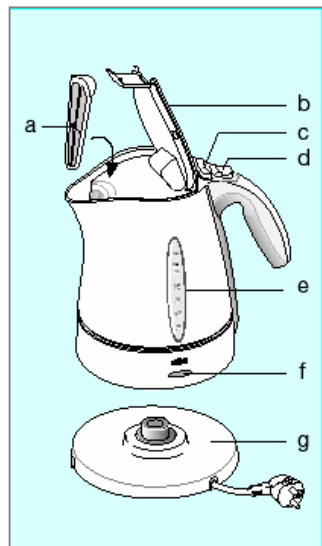
 מחזיר רמה אחת אחורה

 מחזיר לאורך כל רמות הקינון (עד לרמה התחתונה)

דוגמא 7: תרשים מצבים עבור הזמנה



תרגיל כיתה:



- b Lid
- c Lid release button
- d On/off switch
- e Water level indicator
- f Pilot light
- g Base with cord storage

להלן הוראות הפעלה של קומקום חשמלי:

1. מילוי הקומקום: לחצו על לחצן שחרור המכסה (c), פתחו את המכסה (b), מלאו מים וסגרו מחדש את המכסה (הקומקום אינו פועל כאשר המכסה פתוח).
2. הציבו את הקומקום על הבסיס (g).
3. לחצו על מתג ההפעלה (d), התרמוסטט (הנמצא בתוך הקומקום) יחל לפעול ונורית הבקרה (f) תדלק.
4. טמפרטורת המים נבדקת כל 30 שניות. לאחר שהטמפרטורה הגיעה ל- 100 מעלות, יכבה הקומקום באופן אוטומטי ונורית הבקרה (f) תכבה.

5. באפשרותכם לעצור את תהליך החימום בכל רגע על ידי לחיצה נוספת על מתג ההפעלה (d).

הקומקום נכבה אוטומטית בכל אחד מהמקרים הבאים:

- ☒ המים רתחו, כלומר התרמוסטט נכבה.
- ☒ הקומקום ריק ממים.
- ☒ המכסה פתוח.
- ☒ הקומקום הוסר מן הבסיס.

בכל רגע ניתן לנתק את הקומקום מהחשמל ולחברו בחזרה.

עליכם לתאר את מצבי הקומקום ב- Statechart.

הניחו כי מצב התחלתי שלו הוא כאשר הקומקום בחשמל, מונח על הבסיס, ריק, כבוי (כולל נורית בקרה ותרמוסטט, שגם הם כבויים) והמכסה סגור.