



המחלקה להוראת המדעים

סביבת ויזואליזציה JFLAP – פעילויות במודלים חישוביים

מדריך למורה

המחבר: שמואל שוורץ

ייעוץ מדעי: פרופ' מרדכי בן-ארי

תוכן עניינים

(הערך המוצג בסוגריים מציין את מספר דפי הפעילות המוצגים בתחום)

2 מבוא	▪
3 התקנת התוכנה והפעלתה	▪
3 פעילות באוטומט סופי	▪
4 יצירת אוטומט ועריכתו (2)	•
8 תפריטים	•
9 הרצת האוטומט לקלט נתון (3)	○
12 פעולות מתקדמות באוטומט (4)	○
18 פעילות באוטומט מחסנית	▪
18 יצירת אוטומט, עריכתו והרצתו (3)	•
23 פעילות במכונת טיורינג	▪
23 יצירת מכונת טיורינג, עריכתה והרצתה (2)	•
26 נספחים	▪
27 הוראות הורדה והתקנה של סביבת העבודה JRE ו תוכנת JFLAP ...	•
28 בעיה שהתגלתה במהלך ההתנסות	•
29 אינדקס פקודות	•

תכנת JFLAP היא תכנת ויזואליזציה של אוטומטים מסוגים שונים. שימוש נכון בכלי ויזואליזציה טובים יוצר קישור ברור בין עולם המושגים הנלמד לדימוי הקונקרטי המוצג ומקל על התלמידים ליצור מבנים מנטלים. מבנים מנטלים נכונים מסייעים לתלמיד להבין את המתרחש בעולם האמיתי. הויזואליזציה האינטואיטיבית כאשר עוסקים באוטומטים היא שרטוט של האוטומט. השימוש בתכנת JFLAP מאפשר לתלמיד לשרטט את האוטומט ולשמור אותו בקובץ מחשב. השימוש בתכנה מאפשר למורה ולתלמיד להנות ממגוון אפשרויות המסייעות לתלמיד בתהליך הלמידה. מעבר ליכולת השמירה של קבצים יש לתכנת JFLAP מגוון יכולות המאפשרות לתלמיד להשלים מחזור למידה שלם החל מתרגום דרישות מילוליות לאוטומט, דרך בדיקת האוטומט למחרוזות קלט מייצגות וכלה בבדיקת שקילות פתרונות של מספר תלמידים.

כל פרק במדריך זה מציג חלק מתוכנת JFLAP בשני רבדים. הראשון הוא הרובד התכני/פדגוגי המציג את היעדים של כל דף פעילות בתחום התוכן של יחידת הלימוד מודלים חישוביים ודגשים לכל דף פעילות. הרובד השני, הרובד הטכני של תפעול התכנה יוצג כדף הפעלה ללימוד עצמי. הדף בנוי כך שניתן לחלק אותו לתלמידים בלא שינוי במהלך ההוראה תוך שהוא משלב הוראת תוכן עם הוראת התוכנה JFLAP. המורה מוזמן להוריד את קבצי הפעילות מהאתר התומך לאוגדן זה ולשנות ולהחליף את הדפסות המסך וליצור דף דומה על בסיס אוטומטים המועדפים עליו.

התכנים במדריך מוצגים בגישה הויזואלית דהינו ההסברים המילוליים מלווים בהדפסות מסך רלוונטיות. לכל התכנים המוצגים במדריך, מתלווה פסקה המציעה ומדגימה ביצוע של הפעולה או מספר פעולות כדי שהמורה ו/או התלמיד יודא הבנתו לקראת הפעולות שיוצגו בהמשך.

כדי להדגיש את ההמלצה לביצוע, מופיעות המילים "נסה זאת" בתחילת כל פסקה לביצוע והפסקה תחומה על ידי גבול שורה מימין כמו הקו שמימין לשורות אלו.

בסוף המדריך מצורף אינדקס של כל פקודות JFLAP שהוצגו במדריך.

התקנת התכנה והפעלתה

תוכנת JFLAP כתובה בגיזוה ודורשת סביבת הפעלה (JRE) Java Runtime Environment. התקנת סביבת ההפעלה דורשת הפעלה של קובץ התקנה אותו ניתן להוריד מהרשת. גם את תוכנת JFLAP ניתן להוריד מהרשת.

הוראות מפורטות כיצד להוריד את שני הקבצים ניתן למצוא בנספח התקנה במדריך זה.

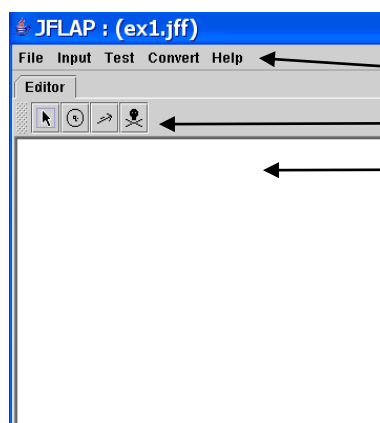


לאחר שסביבת העבודה הותקנה אין צורך בהתקנה נוספת, יש ללחוץ לחיצה כפלה על צלמית הקובץ בשם JFLAP.jar. יתקבל התפריט המוצג משמאל. כפי שניתן לראות זו תוכנה עשירה בכלים ובמדריך זה נציג רק את שלושת הכלים הראשונים: אוטומט סופי, אוטומט מחסנית ומכונת טיורינג.

מומלץ לעקוב אחר הכלים המוצגים במדריך על פי סדר הצגתם שכן ישנן פעולות משותפות לשלושת הכלים. פעולות שכבר תוארו בהרחבה בכלי אחד לא יתוארו שוב בכלי שיוצג אחריו.

הכלי הראשון שיוצג הוא האוטומט הסופי. להפעלה יש ללחוץ על Finite Automaton בתפריט.

פעולות באוטומט סופי



לפניכם חלק ממסך התוכנה לאחר הפעלת כלי האוטומט.

החלון מחולק לשלושה אזורים עיקריים, התפריטים,

לחצני עריכה ליצירת אוטומט

ומסך התצוגה.

בנוסף ישנן פעולות שניתן להפעיל בעזרת

לחיצה על הלחצן הימני בעכבר על אובייקט.

אפשרויות אלו משתנות בהתאם למיקום הלחיצה.

יצירת אוטומט ועריכתו

הערות לדף הפעילות "הכרות עם העורך ב JFLAP" חלקים א' ו ב'

בחלק זה נעסוק ביצירת אוטומט בעזרת התוכנה החל ממסך ריק וכלה בשמירת קובץ אוטומט מלא, תוך שימוש בכל אפשרויות העריכה.

תוכן

בשני דפי פעילות אלו העיקר הוא החלק הטכני. המטרה ליצור אצל התלמיד תחושה שהוא שולט בכלי ושכאשר יתבקש לממש בעזרתו אוטומטים מורכבים הוא ישקיע זמן בעיקר במחשבה הקשורה למה שלמד ולא בחלק הטכני של השימוש בתכנה. עם זאת ישנו גם תוכן מסוים שנספח לתרגיל. האוטומט שהתלמיד מתבקש לשרטט כתרגיל מסכם הוא אוטומט סופי דטרמיניסטי המתאים לדוגמא המוצגת בספר הלמוד של היחידה איור 2.9.

פדגוגיה

חשוב לזכור שזהו מפגש ראשון של התלמידים עם התוכנה. יש להפגיש את התלמיד עם התוכנה לאחר שכבר התנסה בציוור אוטומטים בסיסיי על נייר. הפעילות בזוגות מומלצת למטרת חשיבה, פתרון בעיות ודיון בין התלמידים.

חשוב לבקש מהתלמידים לשמור את הקובץ באופן רציף במהלך הפעילות, לגיבוי ושימוש עתידי.

דפי הפעילות הראשונים פשוטים למדי. ישנם שימושים רבים שאפשריים ולא הוצגו במדריך זה. למשל האפשרות של הכנה מראש של קובץ ובו תוויות מוסתרות. התלמיד יתבקש לפענח מה השפה המוגדרת באוטומט על ידי מעבר בין מצבים והגדרה מה יכולה להיות השפה רק על בסיס במעברים עד למצב מסוים. לאחר דיון בכיתה ניתן להציג את התוויות שהוסתרו (Display State Labels). בשלב הבא תלמיד יעבוד עצמאית מול המחשב ויבדוק את עצמו או תלמיד אחר באותה שיטה. פעילות כגון זו יוצגו באתר המלווה של המדריך.


מטרות

התלמידים ידעו לבנות אוטומטים הזהים לאוטומטים שלמדו בכיתה.

הכרות עם העורך ב JFLAP – חלק א

בדף פעילות זה תלמד ליצור אוטומט ולערוך אותו כרצונך. ביצוע דף פעילות זה יבטיח שתוכל ליצור בתכנה את האוטומטים אותם נהגת לשרטט על נייר עד עתה.


הפעל את התוכנה על פי ההסבר שניתן לך. בתוכנה זו ארבעה כלי עריכה המיוצגים על ידי ארבעה לחצנים. יש לבחור את הכלי המתאים לפני כל פעולה שמבצעים. בחירה של כלי נעשית על ידי לחיצה עליו. לאחר הלחיצה ישתנה הגוון של לחצן הכלי לאפור כדי שיהיה ברור שהוא הכלי הנבחר.

כלי העריכה הראשון שנעסוק בו מיוצג על ידי הלחצן השני משמאל . הכלי מאפשר ליצור מצבים באוטומט. לאחר הבחירה, כל לחיצה על מסך התצוגה תיצור מצב. (לחיצה משמעה לחיצה על הלחצן השמאלי בעכבר ושחרורו). המצבים ממוספרים החל מהמספר 0 לפי סדר יצירתם.

נסה זאת – בחר בכלי יצירת המצבים ולחץ עם העכבר במקומות שונים במסך התצוגה. קבל את המצב המוצג משמאל:



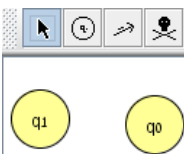
שים לב שלחיצה על מצב קיים אינה משנה אותו אלא ממקמת מצב חדש מעליו.

כלי העריכה השני הוא כלי המחיקה. הלחצן שלו הוא הימני מבין הלחצנים . לאחר הבחירה בכלי זה, לחיצה על אובייקט במסך התצוגה תמחק אותו.

נסה זאת – בחר בכלי המחיקה ולחץ עם העכבר מצבים כדי להישאר עם המצבים q_0 ו q_1 בלבד.

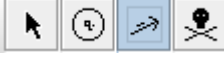


כלי העריכה השלישי הוא לחצן הבחירה. אחת הפעולות האפשריות בלחצן זה היא הזזת מצב שיצרנו ממקום למקום על מסך התצוגה. כדי להזיז מצב יש ללחוץ עליו בעזרת הלחצן השמאלי בעכבר לחיצה רציפה, לגרור אותו למקום הרצוי במסך ולשחרר את הלחצן בעכבר.

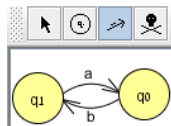


נסה זאת – סדר את המצבים q_0 ו q_1 כמוצג משמאל:

הכרות עם העורך ב JFLAP – חלק א - המשך

כלי העריכה השני מימין  נועד ליצירת מעברים בין מצבי האוטומט. גם הפעם לחיצה על הלחצן בוחרת בו. כדי ליצור את המעבר יש ללחוץ על מצב המוצא לחיצה רציפה, לגרור את העכבר תוך כדי לחיצה אל מצב היעד ולשחרר. בשלב זה נפתחת חלונית בה יש להקליד תו בודד שמיצג את תו המעבר, לאחר ההקלדה יש ללחוץ על לחצן ENTER. (לחיצה על Esc בשלב זה תצא ממצב הכתיבה ותשאיר מעבר "חפשי". בשלב זה יש להימנע ממעברים כאלו ולערוך אותם או למחוק אותם.)

שינוי התו מתבצע על ידי מעבר ללחצן הבחירה . לחיצה כפולה על התו. תיפתח חלונית ובה התו שאינו רצוי, נמחק אותו, נקליד תו ונלחץ עם העכבר על מקום כלשהו במסך.



נסה זאת – הוסף מעברים למצבים q_1 ו q_0 כמוצג משמאל:

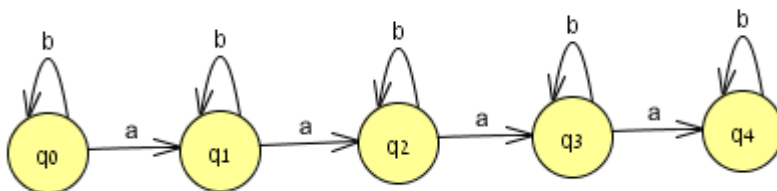
שים לב שניתן למחוק מעבר בעזרת לחצן המחיקה כפי שתואר לעיל.



כדי ליצור מספר אפשרויות מעבר ממצב למצב יש צורך ליצור את כל המעברים במפורש. התוכנה תציג את התווים השונים זה מעל זה כפי שמוצג במסך משמאל: כדי ליצור את המעבר ממצב ל"עצמו" יש ללחוץ על המצב ולגרור את העכבר אל מחוץ למצב ובחזרה אליו.

נסה זאת – תרגיל מסכם – יצירת אוטומט ועריכתו – חלק א

צור את האוטומט המוצג בהמשך:



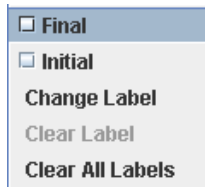
מצב q_0 הוא מצב התחלתי ו q_3 מצב מקבל, ענה לעצמך על השאלה, איזו שפה מייצג אוטומט זה.

כרגע אינך יודע להגדיר מצב כמקבל או כהתחלתי. שמור אוטומט זה להמשך עבודה עליו.

הכרות עם העורך ב JFLAP – חלק ב

פעולות לחצן ימני בעכבר

בתרגיל הסיכום של דף הפעילות הקודם צוין שמצב q_3 הוא מצב מקבל. כדי לסמן מצב מסוים כמצב מקבל יש ללחוץ על המצב בעזרת הלחצן הימני בעכבר. לחיצה על הלחצן הימני על אובייקטים שונים בתכנה תציג אפשרויות שונות לבחירה בהתאם לאובייקט הנלחץ. כדי שהתפריט יוצג יש לבחור תחילה את כלי הבחירה בסרגל הכלים.



לחיצה עם הלחצן הימני על מצב תציג את התפריט המוצג משמאל:

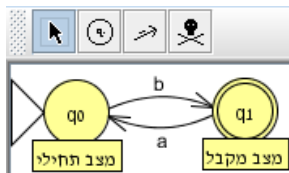
בחירה ב Final תגדיר את המצב כמצב מקבל. בחירה ב Initial תגדיר את המצב כמצב תחילי.

(אם כבר מוגדר מצב אחר כמצב תחילי הוא יאבד תכונה זו.)

בחירה ב Change Label תאפשר לנו להוסיף תווית המתארת את המצב או לשנות תווית קיימת.

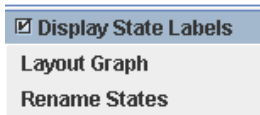
האפשרות Clear Label פעילה כאשר קיימת תווית למצב זה ומאפשרת מחיקתה.

בחירה ב Clear All Labels תנקה את כל התוויות.



נסה זאת – צור בשלבים את האוטומט המוצג משמאל:

צור והגדר מצבים, צור את המעברים, צור תווית אחת ומחק אותה, צור את התוויות הרצויות, הגדר את q_1 כמצב התחלתי ואז את q_0 .



לחיצה על אזור ריק במסך תציג את התפריט המוצג משמאל:

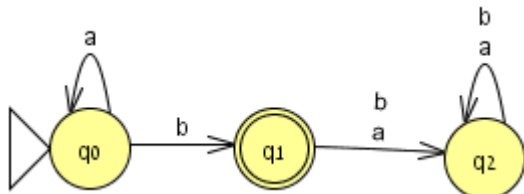
תיבת הבחירה ליד Display State Labels מאפשרת להציג או להסתיר

את התוויות שנכתבו.

האפשרות Layout Graph תארגן את האוטומט על מסך התצוגה בצורה המיטבית.

האפשרות Rename States תמספר את המצבים השונים ברצף. פעולה זו יעילה לאחר מחיקת מצב

כדי לשמור על רציפות מספרי המצבים.



נסה זאת – לסיכום, ולקראת המשך הלימוד צור

את האוטומט המוצג משמאל:

תפריטים

בחלק זה נעסוק בהוראות הקימות בתפריטים השונים. נתחיל בצורות שונות של הוראות הרצה של האוטומט לקלט נתון (3 פעילויות) ונסיים בפעולות מתקדמות שניתן לבצע לבדיקת ושיפור ההבנה.

הערות לדפי הפעילות "הרצת האוטומט לקלט נתון".

תוכן

למעשה בדפי פעילות אלו הלומד משלים את שליטתו בתוכנה ולומד כיצד לבדוק בשיטות שונות קלטים מייצגים לאוטומטים שבנה. אין בחלק זה תכנים חדשים ללמידה ואין שוני מממשי בין הגדרות התוכנה להגדרות בספר הלימוד של היחידה.

פדגוגיה

ישנן שלוש שיטות עיקריות להרצת האוטומטים והן מוצגות בדפי הפעילות השונים. לכל שיטה יתרון יחסי ומומלץ לעודד את התלמידים להשתמש בשיטות שונות ובפרט בשיטת צעד אחר צעד שכן היא מאפשרת הסתכלות והבנה כיצד התהליכים מתרחשים תוך כדי ריצה.

כדאי לבצע תחילה הדגמה בכיתה כך שכולם יראו את אופן פרישת התפריטים וההרצות השונות ורק לאחר מכן יתנסו בכך בעצמם.

חשוב לבקש מהתלמידים לשער מה יקרה. למעשה יש להדגיש שהתוכנה היא רק אמצעי בקרה ובדיקה עצמית ולא תחליף לחשיבה. בפעילויות ההמרה בהן יש אפשרות לעבודה אוטומטית או עבודה ידנית חשוב להדגים ולדרוש מהם שימוש באופציה הידנית כדי להבין כיצד ההמרה מתבצעת.

התפריטים בתוכנה יוצגו על פי הסדר המתאים לדעתנו להכירם ולא בהכרח על פי סדר הופעתם. מומלץ למורה לתכנן מראש כיצד להציג לתלמידים את הפעולות עם התוכנה תוך דגשים על נושאים הקשים להם ובפרט על אי-דטרמיניזם וכיצד יש להבין את ריבוי הנתונים המוצג. כמו כן חשוב שכל מורה ישקול כיצד ואם בכלל להציג את האפשרות Freeze כדי להתמקד בנתיב מסוים של ריצה. מומלץ להדגיש לתלמידים פרטים טכניים שהמורה גילה בעצמו לפני שלב ההצגה בכיתה. כמו למשל שכאשר לוחצים על מצב סיום כדי להציג את טבלת המעברים המתאימה לו הרי שלחיצה זו אינה מבטלת בחירה קודמת ויתכן ויוצגו שתי טבלאות מעברים. דוגמה נוספת היא העובדה שבמדריך לא מוסבר כיצד לסמן מצבים לא דטרמיניסטיים והחלטה האם להדגים זאת היא של המורה.

מטרות

התלמידים ידעו להריץ את האוטומטים שבנו ולהסביר את השלבים השונים בהרצה.

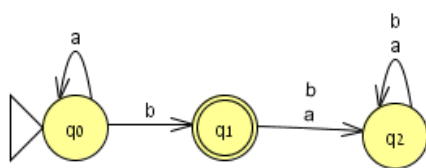
הרצת אוטומט לקלט נתון – חלק א (Fast Run)

לחיצה על תפריט פותחת אותו. אם תחליט לא לבחור אפשרות מהתפריט ניתן לסגור אותו על ידי לחיצה בכל מקום במסך התצוגה או על לחצן Esc במקלדת.



תפריט File הוא תפריט כמעט סטנדרטי בכל תוכנת חלונות ולכן לא נרחיב. האפשרות Dismiss Tab המוצגת באפור אינה פעילה בשלב זה ותידון בהמשך.

נסה זאת – להמשך העבודה עליך ליצור את האוטומט המוצג למטה:



(אם מימשת את ההמלצות הקודמות הוא כבר מוכן)

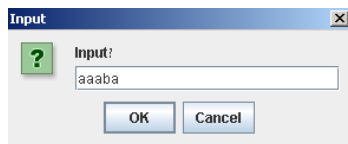
שמור את האוטומט בשם example1.jff.

מיד נבדוק מספר מחרוזות קלט לאוטומט זה.

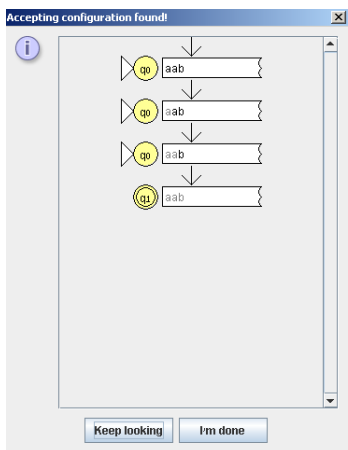
הרצת אוטומט לקלט נתון



תפריט Input הוא תפריט המאפשר בדיקת מחרוזות קלט. ישנן מספר אפשרויות בדיקה וכולן פותחות חלון חדש בו מתבצעת ההרצה. ההבדלים ביניהן הם בתצוגה.

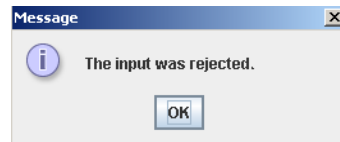


הבדיקה הפשוטה ביותר היא ב Fast Run. לחיצה על אפשרות זו תפתח חלונית בה יש להקליד את מחרוזת הקלט, ולאשר.



במקרה בו המחרוזת תתקבל יוצג נתיב המעברים עד לקבלת המחרוזת. לחצן I'm done יסגור את המסך.

אם המחרוזת נדחית תופיע הודעה בהתאם.



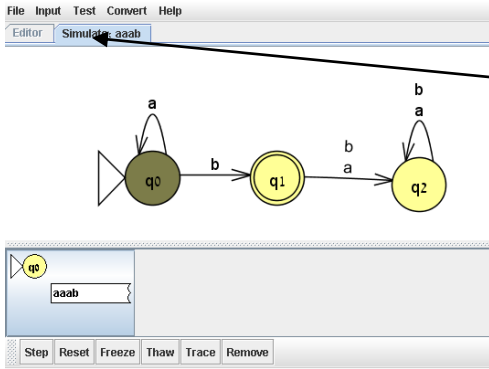
נסה זאת – לפניך שלושה קלטים, שער מי מהם יתקבל באוטומט ובדוק את עצמך בעזרת האוטומט

ששמרת בקובץ example1.jff

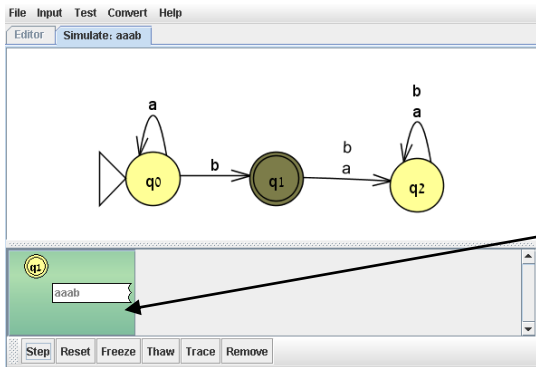
א. aaabbb. ב. aaab. ג. aaaba. ד. aab. ה. ab.

הרצת אוטומט לקלט נתון – חלק ב (Step by State)

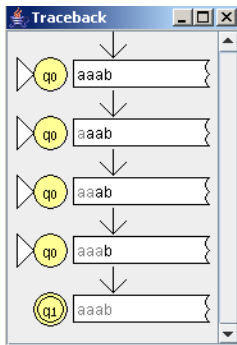
שתי בדיקות מורכבת יותר הן הבדיקות Step by State ו Step with Closure. נציג כעת רק את האפשרות של Step by State. בחירה באפשרות זו מציגה את חלונית הקלט ולאחר הקלדת המחזוריות מוצג חלון חדש בדומה לזה המוצג:



חשוב לשים לב שאנחנו באותו הקובץ אבל יצאנו מהעורך ועברנו לסימולציה של הקלט שהקלדנו. באזור התצוגה מודגש המצב בו אנחנו נמצאים כעת, כלומר המצב התחילי. בחלק התחתון של המסך מתבצעת הבדיקה עצמה. (בדף הפעילות הבא נדון כיצד לחזור לעורך)



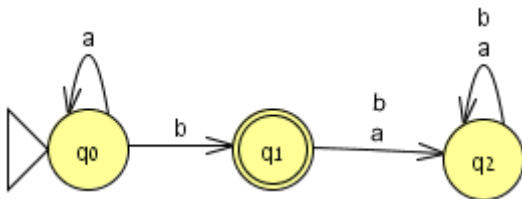
לחיצה על לחצן Step תבצע מעבר אחד. התצוגה תעמעם את התו האחרון שנקלט מהמחזוריות ותציג את המצב העכשווי באוטומט על ידי סימונו בצבע שונה. שם המצב יוצג גם באזור תצוגת המעבר על הקלט. בסיום המעבר על כל המחזוריות, אזור תצוגת המעבר על הקלט יצבע בגוון ירוק אם המחזוריות התקבלה ובגוון אדום אם נדחתה.



ניתן לצפות בכל שלב בהרצה בנתיב עד כה ובפרט בנתיב אל המצב הסופי. לצורך כך יש ללחוץ על אזור הקלט וללחוץ על Trace. מתקבל מסך דומה למסך המוצג משמאל:

לחיצה על Reset תאפשר הרצה נוספת.

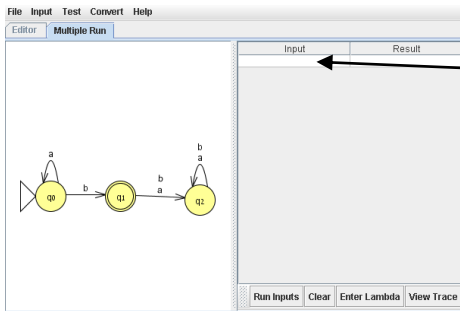
נסה זאת – בדוק את האוטומט הבא לקלט aaaaaaab. האם הגעת למצב מקבל.



הרצת אוטומט לקלט נתון – חלק ג (Multiple Run)

שים לב שבשלב זה אם תלחץ על תפריט File תראה שהפקודה Dismiss Tab מוצגת בשחור. בחירה בפקודה זו תסגור את החלון הנוכחי ותחזירנו למצב העריכה.

בחירה ב Multiple Run מציגה את החלון הבא :



בחלון זה ניתן להציג מספר מחרוזות לבדיקה שיבדקו במקביל. יש להקליד את במחרוזת הראשונה בשורת הקלט ובסיום ההקלדה ללחוץ ENTER. נפתחת שורה חדשה ואפשר להכניס קלט חדש.

בסיום הכנסת כל הקלטים יש ללחוץ על Run Inputs.

Input	Result
a	Reject
aa	Reject
ab	Accept
aab	Accept
abb	Reject
aabb	Reject
aaab	Accept
aaabb	Reject
aaabbb	Reject
aaaba	Reject
aaabba	Reject

בשלב זה יוצגו בסמוך לכל קלט תוצאת הרצתו.

לחיצה על Clear תמחק את כל הקלטים והתוצאות.

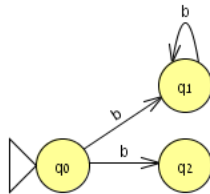
לחיצה על אחד הקלטים ואז על View Trace תציג את מסלול הריצה של הקלט הנבחר.

נסה זאת – לפניך שלוש השערות שגויות שתלמידים העלו לגבי השפה שאוטומט זה מקבל. תכנן לכל

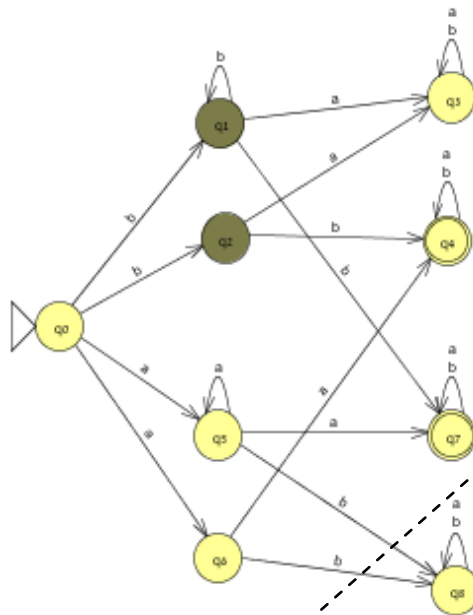
השערה מחרוזת קלט שתוכיח שההשערה שגויה ובדוק אותן במקביל.

1. האוטומט יקבל כל מילה שיש בה לפחות a אחד ובדיוק b אחד.
2. האוטומט יקבל כל מילה שיש בה בדיוק a אחד ולפחות b אחד.
3. האוטומט יקבל כל מילה שיש בה לפחות a אחד ולפחות b אחד.

פעולות מתקדמות באוטומט – חלק א (אוטומט לא דטרמיניסטי)



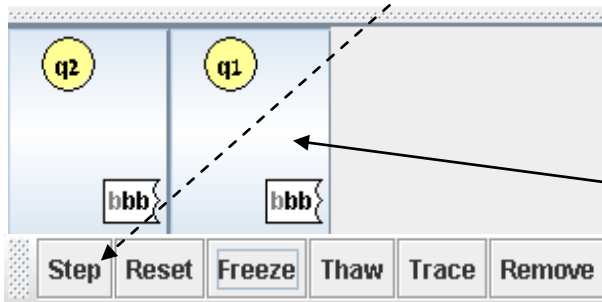
אוטומט המכיל מעברים בדומה למוצג משמאל הוא אוטומט לא-דטרמיניסטי. המעקב אחר הקלט באוטומטים אלו חייב להתבצע בצורה שונה שכן לקלט נתון במצב אחד ניתן לעבור ליותר ממצב אחד באוטומט. המשמעות היא שנדרשת ויזואליזציה שונה.



נסה זאת – בנה את האוטומט המוצג משמאל.

נבצע הרצה לקלט bbb לאוטומט משמאל. נבחר מתפריט Input את Step by State, ונקליד את המחזורות bbb. לחיצה על OK תעביר אותנו למסך סימולציה. בלחצני אזור תצוגת המעבר על הקלט נלחץ על Step, ויתבצע צעד אחד שיציג את המצב המוצג משמאל. המשמעות היא שהאוטומט מבצע במקביל את שני הנתיבים האפשריים לקלט מסוים.

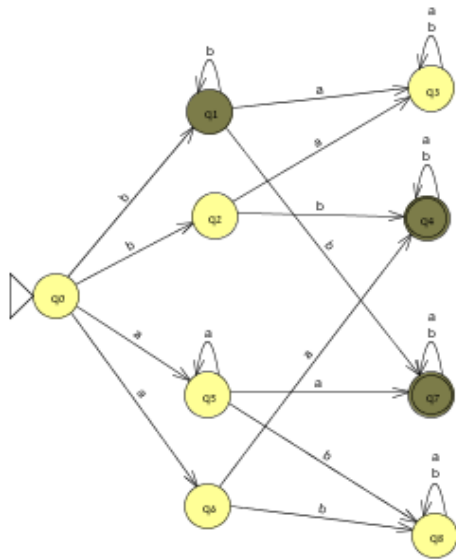
העובדה שהאוטומט אינו דטרמיניסטי מאפשרת שני מעברים ממצב q_0 לאחר תו הקלט b. לכן הלחיצה על Step מבצעת התקדמות במקביל לשני מצבים, המצבים מסומנים באוטומט. כפי שניתן לראות, באזור תצוגת המעבר על הקלט נפתחים שני נתיבי הרצה במקביל. האחד במצב q_1 והשני ב q_2 .



עיון באוטומט מראה שלתו b שבהמשך הקלט, ניתן להמשיך גם ממצב q_1 (ל q_1 או ל q_7) וגם ממצב q_2 (ל q_4). לכן לחיצה נוספת על Step, תבצע את המעברים באוטומט.

ניתן לבחור את אחד הנתיבים על ידי לחיצה עליו ואז להקפא אותו על ידי לחיצה על Freeze וזאת כדי לצפות בהמשך המעברים מהנתיב השני בלבד. בסיום הרצת הנתיב הרצוי יש להפשיר את הנתיב המוקפא על ידי בחירתו, לחיצה על Thaw והמשך ההרצה על ידי Step.

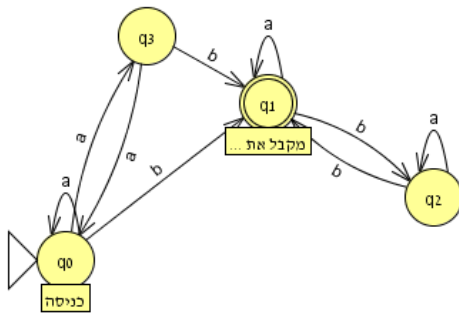
פעולות מתקדמות על אוטומט – חלק א (אוטומט לא דטרמיניסטי) - המשך



- נסה זאת** – בצע את ההרצה שהוצגה כדוגמה.
 כעת שער מה יקרה באזור תצוגת המעבר על הקלט
 למצב q_1 כאשר תלחץ שוב על Step.
 א. מצב q_1 יוחלף במצב q_7 .
 ב. מצב q_1 יעלם ולא יוחלף כי q_7 כבר קיים.
 ג. מצב q_1 יישאר כי הוא אפשרי ויווצר q_7 נוסף.
 ד. מצב q_1 יישאר כי הוא אפשרי אך לא ייווצר q_7 נוסף.
 ה. משהו אחר...
 לחץ על Step לבדוק השערתך. ושוב על Step לסיים.



נסה זאת – בצע הרצה על המחרוזת aaba תוך שימוש בהקפאה של כל נתיב חדש שנפתח פרט לאחד.

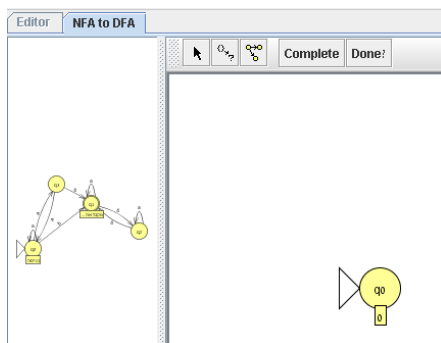
פעולות מתקדמות על אוטומט – חלק ב (Convert to DFA)



לעיתים ניצור להמיר אוטומט שאינו דטרמיניסטי לאוטומט דטרמיניסטי. משמאל מוצג אוטומט לא דטרמיניסטי. ממצב q_0 ישנם שני מעברים אפשריים לקלט a.

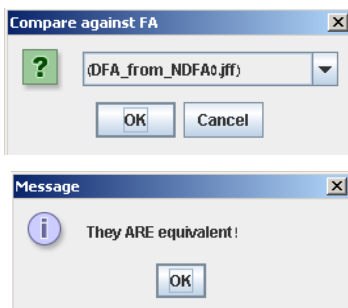
נסה זאת – בנה את האוטומט המוצג ושמור את הקובץ בשם `ndfa_to_dfa0.jff` ומיד נמיר אותו.

כדי להמיר את האוטומט לאוטומט דטרמיניסטי כלומר לאוטומט שמכל מצב בו יוצא רק מעבר אחד לתו קלט נבחר מתפריט Convert את האפשרות Convert to DFA. יפתח מסך חדש (לשונית). בשלב זה בפנינו שתי אפשרויות. הראשונה להשתמש בתוכנה לביצוע ההמרה על ידי לחיצה על Complete. השנייה לבנות את האוטומט המתאים בעצמנו על ידי שימוש בלחצן פרישת המצבים  ו/או בלחצן יצירת מעברים לפי תו קלט  כדי ללמוד מהתהליך עצמו. בשלב זה נסתפק בשיטה הראשונה. לאחר הלחיצה על Complete נותר לחוץ על Done והאוטומט יוצג בקובץ חדש כדי שלא נאבד את ההמרה לאחר סגירת הקובץ של האוטומט המקורי.



נסה זאת – בחר בתפריט Convert את ההמרה לאוטומט דטרמיניסטי ולחץ על Complete. סדר את המצבים כך שתוכל להשוות בקלות את האוטומט שהתקבל לאוטומט המקורי. שער מה משמעות התוויות שמוצמדות לכל מצב. חץ על Done וסדר את האוטומט שנוצר כך שיהיה קל להשוות בינו לבין המקור הלא דטרמיניסטי. שמור את הקובץ בשם `dfa_from_ndfa0.jff`

השוואת האוטומטים השונים עשויה לעזור לשיפור ההבנה כיצד לממש באופן דטרמיניסטי מעברים שאנו חושבים עליהם כאפשריים במקביל. זה זמן טוב להכיר פעולה ראשונה בתפריט Test, הפעולה Compare Equivalence. הפעולה משווה שני אוטומטים. נשתמש בה להשוות את האוטומט



המקורי – הלא דטרמיניסטי לזה החדש – הדטרמיניסטי. בחירת פעולה זו בתפריט מציגה חלונית המאפשרת בחירה בקובץ אליו רוצים להשוות את האוטומט הנתון. לחיצה על OK תציג את התשובה. במקרה שלנו ברור מדוע היא חיובית.

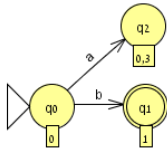
פעולות מתקדמות על אוטומט – חלק ג (המשך Convert)


ראינו שניתן לבצע את פעולת ההמרה בצורה אוטומטית. חשוב לבצע את פעולת ההמרה לפחות מספר פעמים באופן "ידני" שכן התשובה המיידית אינה תורמת להבנה כיצד הדבר מתבצע.

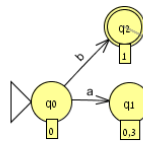
בחלק הימני של מסך התוכנה מופיע המצב q_0 אשר כמו באוטומט המקור מוגדר כמצב תחילי. התווית המצורפת היא מספר המצב שהוא מייצג באוטומט המקור.



הצורה הקלה לפרוש את המצבים של האוטומט החדש היא לבחור בלחצן פרישת המצבים ולאחר מכן ללחץ על מצב קיים. התוצאה תהיה הצגה של כל המעברים ממצב זה ואת המצבים אליהם הם עוברים. היתרון בשיטה זו על השיטה הקודמת היא שיש לנו שהות לשער מה יהיו המעברים שיוצרו. במקרה שלנו נוצרו שני מצבים חדשים.



שיטה טובה יותר היא להשתמש בלחצן יצירת המעברים , כלי זה מאפשר לנו ליצור באופן פעיל את המעברים והמצבים. להבדיל מהעורך ניתן כאן ליצור מעבר למקום ריק במסך והמצב יוצר לאחר מכן.




נלחץ על הלחצן ונגרור קו ממצב קיים לאזור ריק במסך או למצב קיים במסך. תיפתח חלונית בה יש להקליד את תו המעבר. לאחר לחיצה על OK, תיפתח חלונית חדשה ובה יש להקליד את מספר המצב שאליו מוביל תו המעבר שהוספנו באוטומט המקורי. אם יש יותר ממספר

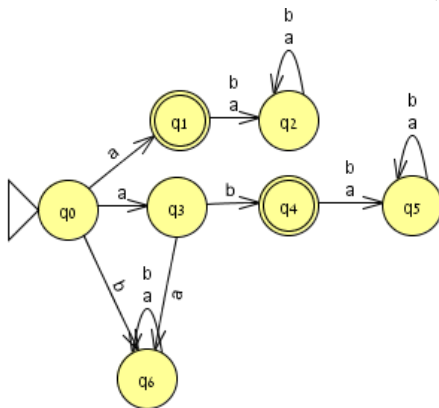



אחד, יש להפריד ביניהם בפסיק. בחירה נכונה תציג מעבר ומצב חדש, שגיאה תציג הודעה מתאימה. ניתן בכל שלב לבדוק האם תהליך הבנייה הסתיים על ידי לחיצה על Done? אם חסרים פריטים תתקבל הודעה כמו זו המוצגת משמאל. כשמסתימת בניית האוטומט מוצגת הודעה מתאימה.



נסה זאת – אם לא ביצעת את התרגול עד כה, פתח את הקובץ `ndfa_to_dfa0.jff` ובחר מתפריט Convert את האפשרות Convert to DFA. אם ביצעת את התרגול הקודם והשתמשת Complete בחר ב `Dismiss Tab` מתפריט File והתחל את התהליך שוב. בשלב זה מוצג לפניך מצב q_0 בלבד. כדי להכין את המסך לעבודה נוחה, גרור את הגבול בין חלקי המסך ימינה כדי להגדיל את תצוגת אוטומט המקור. לחץ על לחצן הבחירה  והזז את מצב q_0 למקום תואם למצב זה באוטומט

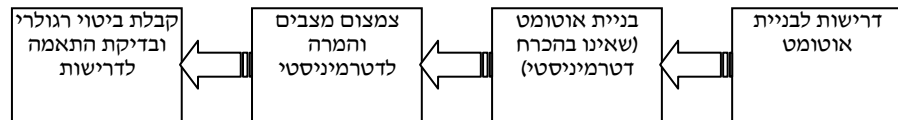
המקור. הוסף את המעבר ממצב q_0 למצב q_1 באוטומט המקור. השלם את האוטומט.



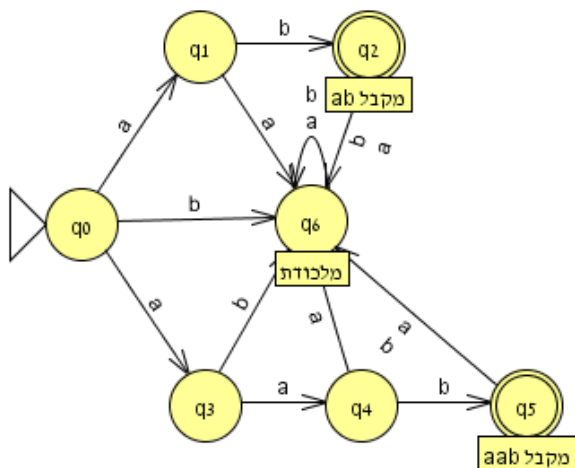
נסה זאת – משמאל מוצג אוטומט שנבנה על מנת לקבל את המחרוזות a ו ab בלבד. קל לראות שהאוטומט אינו דטרמיניסטי ושיש כפילויות מיותרות – המר את האוטומט על ידי שימוש בלחצן יצירת המעברים , לאוטומט דטרמיניסטי.

פעולות מתקדמות על אוטומט – חלק ד (Convert FA to RE)

דף הפעלה זה מציג רצף פעולות אפשרי לפתירת שאלה.




הדרישות לבניית האוטומט: בנה אוטומט סופי דטרמיניסטי המקבל את השפה $L = \{ab, aab\}$. ניסוח אחר: הוכח על ידי בניית אוטומט סופי דטרמיניסטי שמקבל את השפה שהיא שפה רגולרית. התרגיל הוא הדוגמה המתאימה לאיור 3.6 בספר הלימוד של היחידה.



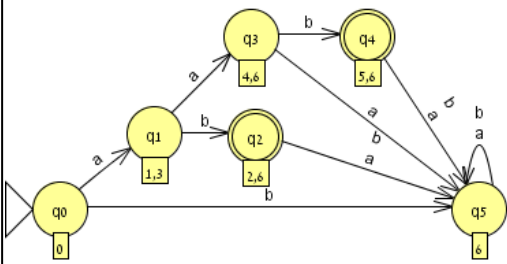
האוטומט עשוי להיראות כמו זה שמשמאל – שים לב שהוא אינו דטרמיניסטי!! יש להמיר אותו לאוטומט דטרמיניסטי סופי כדי להוכיח שהשפה רגולרית.

לפני ההמרה, רצוי לבחור בתפריט Test האפשרות Highlight Nondeterminism, מצב q_6 יצבע בצבע שונה שכן הוא היחיד שאינו דטרמיניסטי.

צמצום מצבים והמרה לדטרמיניסטי. מומלץ לבצע את ההמרה על ידי שימוש בלחצן יצירת המעברים , כדי להבין טוב יותר מדוע ניתן לאחד את מצבים.

נסה זאת – בנה את האוטומט המוצג ולאחר מכן בחר מתפריט Convert את האפשרות Convert to DFA. ובנה בעצמך את האוטומט הדטרמיניסטי השקול.

פעולות מתקדמות על אוטומט – חלק ד (Convert FA to RE) - המשך

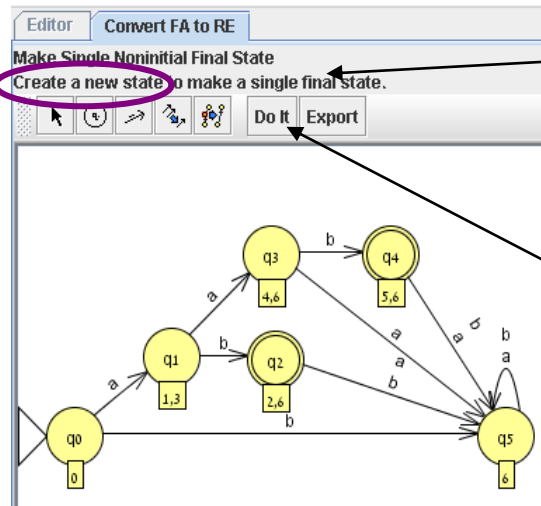


התקבל האוטומט המוצג משמאל. חשוב לשים לב שיתכן והמספרים יהיו שונים שכן כל תלמיד מייצר את המצבים בסדר אחר. אולם התוויות יהיו זהות! כרגע בידנו אוטומט סופי דטרמיניסטי ומכאן שהשפה אותה הוא מיצג היא רגולרית.

חשוב לציין שזהו אינו האוטומט המינימלי שניתן לבנות. פעולת המינימיזציה אפשרית מתפריט Convert ולא נתייחס אליה במדריך זה. נציין שניתן בקלות לראות שמצבים q_2 ו q_4 ניתנים לאיחוד.

קבלת ביטוי רגולרי ובדיקת ההתאמה לדרישות: מתפריט Convert נבחר ב Convert FA to RE

שמשמעו המרה של אוטומט סופי לביטוי רגולרי. מתקבל המסך הבא:

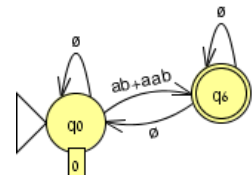


המשך ההמרה נעשה במספר שלבים, יש לפעול על פי ההוראות המופיעות בשורת ההנחיות.

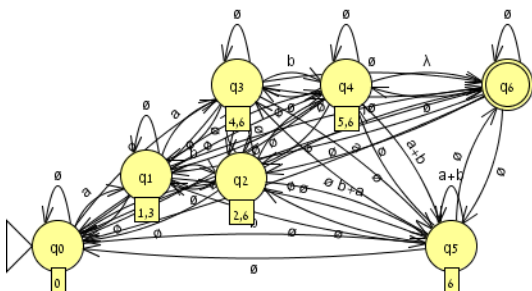
את ההנחיה הראשונה חייבים לבצע באופן ידני כלומר ממש ליצור את המצב המקבל שיחליף את כל המצבים המקבלים באוטומט הקיים. הביצוע הוא על ידי לחיצה על לחצן המצבים ואז לחיצה במקום כלשהו על המסך.

בהוראות בהמשך ניתן ללחוץ על לחצן Do It והתוכנה תבצע את הנדרש בעצמה.

מימין מוצג האוטומט

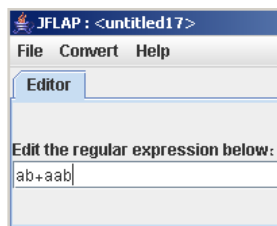


שמתקבל בסוף התהליך ומיצג את השפה L.



נסה זאת – בצע את ההמרה על פי ההוראות.

בתהליך ההמרה התוכנה עושה שימוש באלגוריתם הדורש אוטומט בו כל מצב מקושר לכל מצב אחר לכן באחד משלבי הביניים האוטומט יראה כמוצג משמאל. חשוב להתנסות שכן יש בפעולות הללו תרומה משמעותית להבנת שיטת "מיצוי" השפה מהאוטומט.



סיום התהליך נעשה על ידי לחיצה על לחצן Export. יפתח חלון חדש ובו הביטוי/שפה שהאוטומט שלנו קיבל. שימו לב שיש אפשרות לערוך את הביטוי. פעולת העריכה זו לא תוסבר במדריך זה.

פעילות באוטומט מחסנית



מהתפריט נבחר באוטומט מחסנית. Pushdown Automaton.

המסך שיפתח זהה לזה של אוטומט סופי אולם התפריטים שונים.

יצירת אוטומט, עריכתו והרצתו

הערות לדפי הפעילות " אוטומט מחסנית "

בחלק זה נעסוק ביצירת אוטומט מחסנית, נתחיל ממסך ריק אולם מומלץ להכיר קודם את פעולות העריכה על אוטומט סופי.

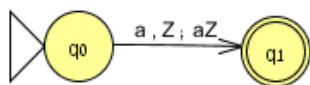
תוכן

אוטומט מחסנית הוא הרחבה של האוטומט הסופי וכוחו בזיכרון שנוסף לו – המחסנית. מבחינת התוכנה מצבי האוטומט נראים זהים אולם המעברים מוגדרים בצורה מורכבת קצת יותר. בדפי הפעילות הבאים נכיר את אוטומט המחסנית תוך דגש מימושו בתוכנה.

פדגוגיה

ההוראה מתבצעת כמו באוטומט הסופי. כמו כן לתלמידים רבים יש קושי עם המעברים הלא דטרמיניסטיים ובעיקר עם התחושה שהאוטומט מחליט בדיעבד באיזה מסלול לבחור.

חשוב לציין שהתוכנה מממשת את פעולת ההצצה למחסנית על ידי הוצאה של האיבר שבראש המחסנית והחזרתו אליה. בתחתית המחסנית איבר דמה שמסומן באות Z .



פעולת ה POP שמחליפה את פעולת ה TOP דורשת מאיתנו להחזיר למחסנית את מה שהוצאנו ובפרט את סמן המחסנית

הריקה. חשוב להדגיש לתלמידים שהשאלה האם המחסנית ריקה היא האם בראש המחסנית נמצא Z וליתר דיוק – האם הוצאנו את Z . ואם כן אז מיד להחזיר אותו פנימה!!!

מטרות

התלמידים ידעו לבנות אוטומט מחסנית ולהסביר את ההבדלים בין אוטומט המחסנית ואוטומט סופי.

התלמידים ידעו להבחין בין אוטומט מחסנית דטרמיניסטי וכזה שאינו דטרמיניסטי ויוכלו לתאר את מהות ההבדל.

יצירת אוטומט מחסנית ועריכתו

בדף פעילות זה תלמד ליצור אוטומט מחסנית ולערוך אותו כרצונך.



למעשה אתה מכיר את כלי העריכה ויש רק להתמקד בשינויים שבכלי המעברים. גם הפעם לחיצה על הלחצן בוחרת בו. כדי ליצור את המעבר יש ללחוץ על מצב המוצא לחיצה רציפה, לגרור את העכבר תוך כדי לחיצה אל מצב היעד ולשחרר. נפתחת חלונית ובה שלושה חלקים.

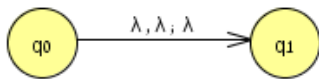


בחלק השמאלי יש להקליד תו בודד שמיצג את תו המעבר (קלט).

בחלק האמצעי יש להקליד את התו שמוצא מהמחסנית לבדיקה.

בחלק הימני יש להקליד את התווים אותם יש להכניס למחסנית אם התו המתאים (שהוקלד בחלק האמצעי) נמצא בראש המחסנית. ניתן לעבור בין התאים על ידי העכבר או בעזרת לחיצה על Tab.

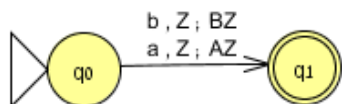
לאחר ההקלדה יש ללחוץ על לחצן ENTER או ללחוץ במקום כלשהו על המסך.



אם לחצנו עם העכבר במקום כלשהו על המסך ולא הקלדנו דבר בחלונית או שלחצנו על Esc התוכנה תצא ממצב הכתיבה ותשאיר מעבר "חפשי". בשלב זה יש להימנע ממעברים כאלו ולערוך אותם או למחוק אותם.

שינוי התווים מתבצע על ידי מעבר לכלי הבחירה , לחיצה כפולה על התו, תיפתח חלונית ובה התו שאינו רצוי, נמחק אותו, נקליד תו ונלחץ עם העכבר על מקום כלשהו במסך.

נסכם את משמעויות המעברים המוצגים באוטומט שמשמאל.



אם תו הקלט הוא b ובמחסנית יש את התו Z יש להכניס למחסנית את צמד התווים BZ.

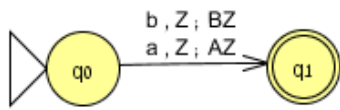
כלומר אם לפני הקלט המחסנית נראתה כך: $\begin{matrix} | \\ | \\ Z \end{matrix}$: כלומר הייתה מחסנית ריקה. אז לאחר הקלט היא תראה כך: $\begin{matrix} | \\ | \\ B \\ | \\ Z \end{matrix}$

אם תו הקלט הוא a ובמחסנית יש את התו Z יש להכניס למחסנית את צמד התווים AZ.

כלומר אם לפני הקלט המחסנית נראתה כך: $\begin{matrix} | \\ | \\ Z \end{matrix}$: כלומר הייתה מחסנית ריקה. אז לאחר הקלט היא תראה כך: $\begin{matrix} | \\ | \\ A \\ | \\ Z \end{matrix}$

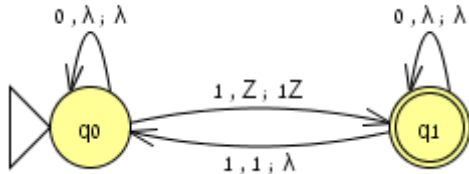
יש לשים לב שהאות Z (אות 'גדולה') מייצגת את תחתית המחסנית. היא נמצאת שם ברגע אתחול המחסנית ואם מוציאים אותה כדי לבדוק מה בראש המחסנית יש להחזירה פנימה. שימוש באות z (אות 'קטנה') יגרור שגיאה.

יצירת אוטומט מחסנית ועריכתו - המשך



נסה זאת – צור את האוטומט המוצג משמאל.

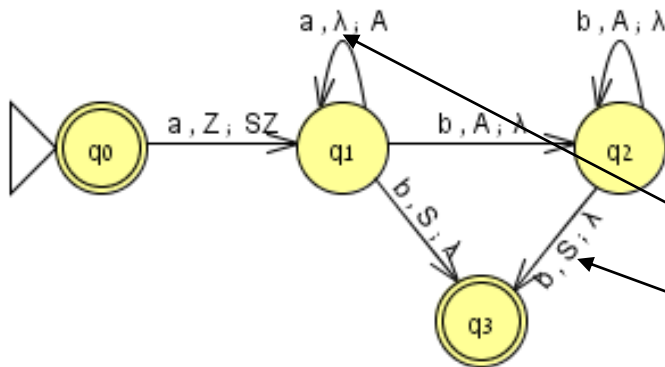
השתמש בפעולות עריכה והוסף מעבר אשר בו לקלט c ולמחסנית ריקה יוכנס CZ למחסנית. שמור את הקובץ.



משמאל מוצג אוטומט מחסנית שמקבל את כל המילים בהן מספר ה 1 -ים הוא אי זוגי (מעל $\{0,1\}$). נדגיש שוב את העובדה שהתוכנה אינה "מציצה" לראש המחסנית אלא מוציאה את האבר שבראשה ויש להכניסו חזרה

פנימה אם צריך לשמור אותו ובפרט כשהמחסנית ריקה – יש צורך להחזיר לפני התו את האות Z . כאשר לא רוצים להחזיר דבר (שכן כבר הוצאנו תו) או שלא רוצים לבדוק דבר, נשאיר את התא המתאים ריק ויתקבל בו התו λ .

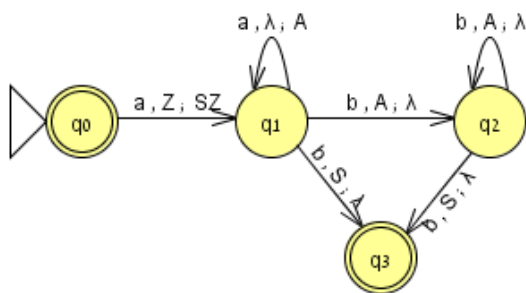
יתכן ושמת לב שהשימוש במחסנית באוטומט זה הוא מנוון שכן אם היא ריקה מוסיפים לה את התו 1 ואם יש בה את התו 1 אז מוציאים אותו. הסיבה לניווך היא העובדה שכדי לזכור אם המספר זוגי או לא מספיק אוטומט "רגיל". אוטומט המחסנית.



נסה זאת – תרגיל מסכם – צור את האוטומט המוצג משמאל. שים לב כיצד כדי להוסיף תו למחסנית לקלט מסוים בלי תלות במה שיש בראשה מוסיפים תו למחסנית בלי לבדוק. בצורה דומה ניתן להוציא תו בשלב הבדיקה ולא להחזיר דבר.

האוטומט מוצג בספר הלימוד של היחידה, איור 5.2.

הרצת אוטומט מחסנית לקלט נתון – אוטומט דטרמיניסטי



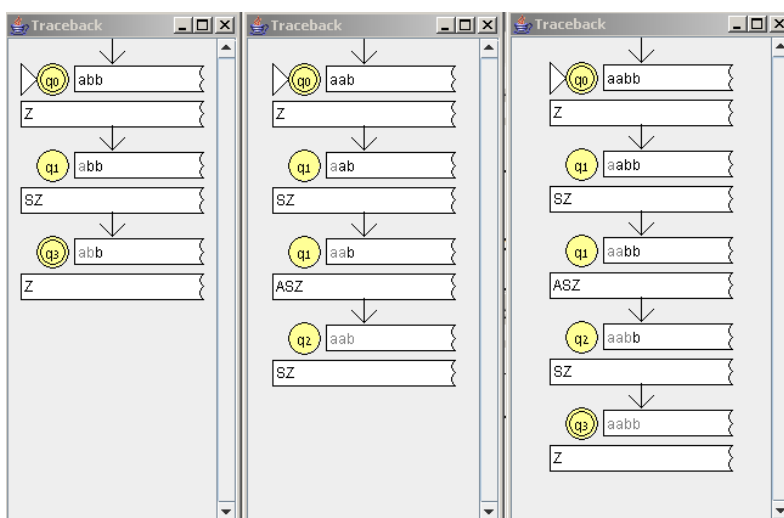
נשתמש באוטומט שיצרנו בדף הפעילות הקודם כדי לבדוק קלטים שונים.

נבדוק את התהליך על שלוש מילים שונות:
 (1) .abb (2) .aab (3) .aabb.

נשתמש בהוראה המוכרת לנו. מתפריט Input

נבחר ב Multiple Run. נקליד את המילים

השונות, נריץ ונציג את המסלולים (View Trace).



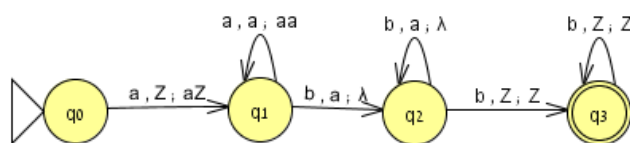
Input	Result
abb	Reject
aab	Reject
aabb	Accept

קיבלנו חשבון לשים לב שלמרות שלכאורה קריאת המילה .abb מסתיימת במצב מקבל למעשה היא אינה מסתיימת שם, ישנו תו קלט נוסף שגורם לכך שהמילה נדחית.

רק כאשר המילה נקראת עד

סופה ומגיעים למצב מקבל אז המילה מתקבלת. האוטומט מקבל את המילים המתחילות ברצף של התו a ולאחריו רצף זהה של התו b. q_0 הוא מצב מקבל כדי להכיל את המילה הריקה (a^0b^0).

נסה זאת – צור את



האוטומט המוצג משמאל. זהו אוטומט המזהה מילים המקיימות חוקיות מסוימת בין מספר תווי ה a שיש בהן ומספר תווי ה b בהן.

תחילה נסה לשער מבלי להריץ את האוטומט ולאחר מכן השתמש בהרצה של מילים ייצוגיות לבדיקת השערתך.

הרצת אוטומט לקלט נתון – אוטומט לא דטרמיניסטי

נבחן מקרה בו קיים אוטומט מחסנית לא דטרמיניסטי המסוגל לקבל שפה שאוטומט מחסנית דטרמיניסטי לא מסוגל לקבל. נבנה אוטומט כזה על מנת לזהות פולינדרומים.

נשתמש בזיהוי של האות הראשונה בצורה מיוחדת. חוץ מ A ו B שמוכנסים בהתאמה לקלט a ו b אנחנו מכניסים S אם האות הראשונה במילה היא a ו T אם האות הראשונה במילה היא b . כמו כן הסימן Z מגדיר את תחתית המחסנית כלומר מחסנית ריקה.

(מקרה זה מוסבר בהרחבה כולל ניסוח פורמלי בדוגמה 5.4 בספר הלימוד של היחידה. בתוכנה המעברים שונים במקצת מאלו שבספר הלימוד עקב השימוש ב Z במקום ב \perp וב λ במקום ב ε).

נסה זאת – השתמש בקבוצת המעברים המוגדרת ונבנה את האוטומט (מוצג בהמשך). בדוק האם הוא מקבל את השפה.

מעברים מהמצב ההתחלתי q_0 :

(q_0, a, Z, q_1, ZS) ; (q_0, b, Z, q_1, ZT) המחסנית ריקה, לכן הוצאנו את Z והכנסנו צמד בהתאמה.

מעברים מהמצב הראשון q_1 : נקראה לפחות אות אחת, ונמצאים עדיין בחלקה הראשון של המילה. הראשון (שלב האגירה) ובאות השנייה.

(q_1, a, S, q_1, SA) ; (q_1, b, T, q_1, TB) ; (q_1, a, T, q_1, TA) ; (q_1, b, S, q_1, SB) המשמעות היא שאנחנו בחלק הראשון (שלב האגירה) ובאות כלשהי.

$(q_1, a, S, q_3, \lambda)$; $(q_1, b, T, q_3, \lambda)$ מעבר זה מניח שבדיוק עברנו לחלק השני (שלב המחיקה) וזה קרה באות השנייה כלומר היא גם הלפני אחרונה.

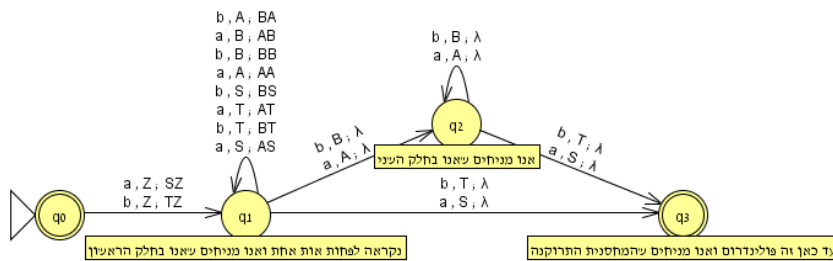
$(q_1, a, A, q_2, \lambda)$; $(q_1, b, B, q_2, \lambda)$ מעבר זה מניח שעברנו לחלק השני (שלב המחיקה) באות כלשהי.

מעברים מהמצב השני q_2 :

$(q_2, a, S, q_3, \lambda)$; $(q_2, b, T, q_3, \lambda)$ אנחנו בחלק השני (שלב המחיקה) באות הלפני אחרונה.

$(q_2, a, A, q_2, \lambda)$; $(q_2, b, B, q_2, \lambda)$ אנחנו בחלק השני (שלב המחיקה) באות כלשהי.

q_3 : עד לכאן קלטנו פולינדרום, אנו מניחים שהקלט הסתיים והמחסנית ריקה.



חשוב להבין שכאשר

נממש את המעבר

בתוכנה נכתוב TZ ולא

ZT כי כאן הצירוף מיצג

את השינוי ובתוכנה את

הצמד להכנסה למחסנית (מוכנסים "יחד" ולא כשני אברים). כך גם נכתוב AS במקום SA וכו.

לאחר שבנית את האוטומט הרץ אותו (Step by Step) לפולינדרום (למשל aaabbaabbaaa) ועקוב

אחרי הריצה הלא דטרמיניסטית. באר לעצמך מה משמעות כל צעד בכל נתיב שנפתח.

פעילות במכונת טיורינג

מהתפריט נבחר ב **Turing Machine**.
הערות לדפי הפעילות "יצירת מכונת טיורינג, עריכתה והרצתה"

Finite Automaton
Pushdown Automaton
Turing Machine
Multi-Tape Turing Machine
Grammar
L-System
Regular Expression

תוכן

מספר פרטים שצריך להדגיש: ראינו עד כה התפתחות ביכולות האוטומט. מגבלתו של האוטומט הסופי היא שלפי הגדרתו מספר המצבים שלו סופי ולכן היכולת ל"זכור" בעזרת מצביו היא סופית. אוטומט המחסנית יכול היה "לזכור" פרטים שהאוטומט הסופי לא יכול היה לזכור בעזרת המחסנית. המחסנית – מהיותה אינסופית – מאפשרת לאוטומט לזכור אירועים שלא ניתן לזכור באמצעות המצבים

בלבד. למשל את השפה $L = \{ a^n b^n \mid n \geq 0 \}$ ניתן לזהות בעזרת אוטומט מחסנית ולא בעזרת אוטומט סופי. עיקר כוחו של אוטומט המחסנית היא גם מגבלתו. על פי הגדרתה המחסנית מאפשרת גישה רק אל ראש המחסנית. מכונת טיורינג משתמשת גם היא בזיכרון אינסופי המוגדר כך שניתן לגשת לכל תא בזיכרון. לאור העובדה שמכונת טיורינג יכולה לטייל הלוך ושוב על סרט הזיכרון הרי שבמכונת טיורינג יתכן ובמכונה נתונה עם קלט נתון אין משמעות למונח "סוף הקלט". במקרה כזה יש רק תנאי עצירה אחד – **כאשר הצעד הבא אינו מוגדר**.
מחרוזת תתקבל אם העצירה היא במצב מקבל ותידחה אם העצירה היא במצב שאינו מקבל.

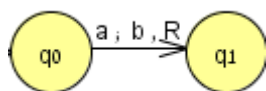
פדגוגיה

בשלב זה התלמידים כבר מומחים בתוכנה ולכן אפשר לתת להם להתנסות במכונת טיורינג לבדם. למרות זאת רצוי לפתור את אחד מהתרגילים האחרונים בפורום כיתתי תוך דגש על כך שיש כאן קפיצת מדרגה מבדיקת סיום הקלט במצבים מקבלים לשימוש במכונה להצפנה או תיקון שגיאות תקשורת.
הבדל חשוב אחד בתוכנה זו לעומת המודל הנלמד בספר הלימוד של היחידה הוא שבתוכנה אין סימן תחילת סרט (\vdash). המשמעות היא שהסרט אינסופי לשני הכיוונים. לצרכי תכנון ומימוש מכונות טיורינג אין בכך מגבלה. כמו כן חשוב לציין שבתוכנה כדי לציין תו ריק Δ – פשוט אין לכתוב דבר והתא יסומן בתו \square .

מטרות

התלמידים ידעו לבנות מכונת טיורינג בדומה לאלו שנבנו בכיתה וכן לזהות על ידי הרצה של קלטים את מטרת המכונה.

יצירת מכונת טיורינג, עריכתה והרצתה



לפניכם דוגמא כיצד נראה ובהתאמה כיצד מוגדר מעבר במכונת טיורינג.

המעבר המוצג משמאל מוגדר כחמישייה: (q_0, a, q_1, b, R)

q_0 – המצב ממנו מתבצע המעבר.

a – תו שאותו המכונה תקרא. התו יכול להיות תו מא"ב הקלט או מא"ב המכונה או Δ או \perp

q_1 – המצב אליו מתבצע המעבר.

b – תו שאותו המכונה תכתוב. התו יכול להיות תו מא"ב הקלט או מא"ב המכונה או Δ או \perp

R – מייצג את המילה ימין כלומר לאחר סיום הקריאה והכתיבה יש להזיז את מצביע הראש ימינה.

(קיימת אפשרות כמובן גם ל L – להזיז שמאלה. בתוכנה יש אפשרות גם ל S – לא להזיז)



נסה זאת – בנה את המכונות המוצגות משמאל האם הן מייצגות

אותה שפה? שים לב ל R ו L ניתנים לבחירה מתיבת גלילה אם

תלחץ על התא המתאים.



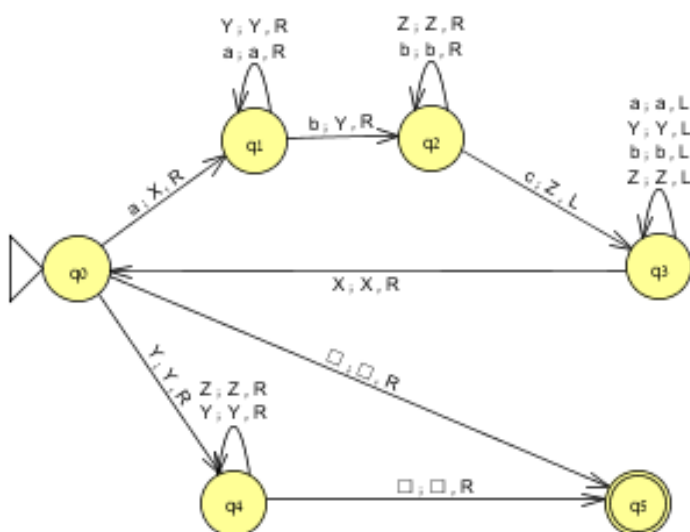
כאשר נריץ קלט למכונת טיורינג יוצג לפנינו קטע סרט ובו ממוקם הקלט בתוך רצף של תאים ריקים.

המכונות המוצגות משמאל למעלה מקבלות את שפת כל המילים אשר

מתחילות בשני תווים של a . המצב המוצג מזכיר מאוד אוטומט סופי אולם

חשוב לציין שאין צורך במלכודות שכן ברגע שהגענו ל q_2 אין משמעות להמשך הקלט לכל תו קלט.

זהו מצב מקבל ואין אף צעד מוגדר מכאן לכן מצב זה עצמו הוא מלכודת לכל קלט.



נסה זאת – משמאל מכונת טיורינג.

רשום מהו א"ב השפה.

רשום מהו א"ב המכונה.

בנה את המכונה בתוכנה ועקוב אחר

מחרוזות קלט מייצגות.

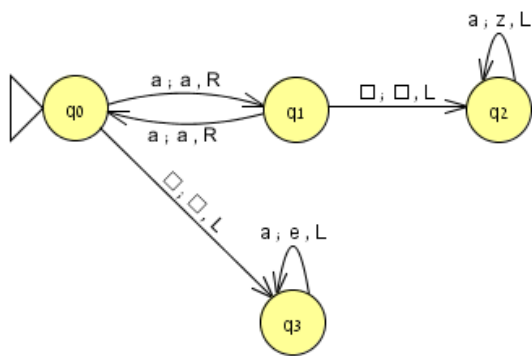
איזו שפה מקבלת המכונה?

מכונה זו זהה לאיור 7.6 בספר הלימוד

של היחידה.

מכונת טיורינג, הרחבת משמעות הסרט

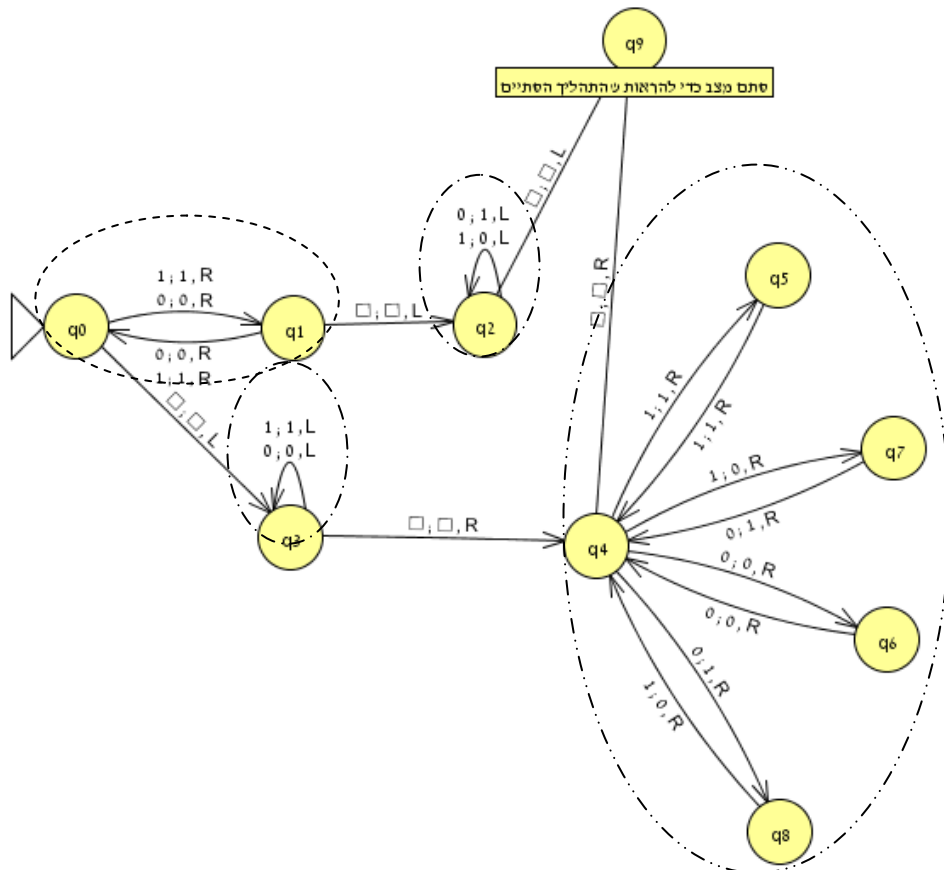
עד כה השתמשנו באוטומט על מנת לבדוק האם מחרוזת מסוימת מתקבלת על ידי האוטומט. חשוב להכיר דוגמה לשימוש נוסף שניתן לעשות במכונת טיורינג מעבר לבדיקת מחרוזות קלט. העובדה שמכונת טיורינג מאפשרת כתיבה על הסרט נותנת משמעות חדשה לתווים שרשומים על הסרט עצמו בסוף התהליך.



נסה זאת – בנה את המכונה המוצגת משמאל

והרץ אותה צעד אחר צעד לקלטים $.aaaa$ ו $.aaaaa$. שים לב לא ללחוץ מספר צעדים גדול מידי כדי שתוכל לראות מה רשום על הסרט בסיום ההרצה. שים לב שבמכונה הזו כלל אין מצב מקבל... מה מבצעת המכונה?

נסה זאת – מכונת טיורינג שאינה דטרמיניסטית יכולה לשמש להצפנה. שים לב לחלקים השונים במכונה שבהמשך. נסה לזהות מה תפקידו של כל חלק ובסופו של תהליך מה מבצעת המכונה. בנה את המכונה ובדוק את השערותיך בהרצת קלטים מיצגים (01, 00, 001, 111, 111000101 וכו').



נספחים

הוראות הורדה והתקנה של סביבת העבודה JRE ותוכנת JFLAP

הורדה והתקנה של סביבת העבודה JRE

1. היכנסו לכתובת <http://java.sun.com/j2se/1.4.2/download.html>.

האתר נראה בערך כך :



The screenshot shows the Java EE SDK download page. It features the NetBeans logo on the left and the Java EE SDK logo on the right. Below the logos, there are two main sections: 'J2SE v 1.4.2_12 SDK' and 'J2SE v 1.4.2_12 JRE'. The 'J2SE v 1.4.2_12 JRE' section has a 'Download J2SE JRE' link highlighted with a red arrow. There are also links for 'Installation Instructions', 'ReadMe', 'ReleaseNotes', 'Sun License', and 'Third Party Licenses' for both sections.

2. לחצו על Download J2SE JRE.

3. תועברו לדף שבחלקו המרכזי השורות הבאות :



The screenshot shows the license agreement page. It features a 'Required' section with a 'You must accept the license agreement to download the product.' message. Below this, there are three radio buttons: 'Accept License Agreement', 'Review License Agreement', and 'Decline License Agreement'. The 'Accept License Agreement' button is selected. Below the radio buttons, there is a table with two rows of download links for Windows Platform - Java(TM) 2 Runtime Environment, Standard Edition 1.4.2_12.

Windows Platform - Java(TM) 2 Runtime Environment, Standard Edition 1.4.2_12	
↓ Windows Offline Installation, Multi-language	j2re-1_4_2_12-windows-i586-p.exe
↓ Windows Installation, Multi-language	j2re-1_4_2_12-windows-i586-p-iftw.exe

4. לחצו על Accept כך שתיווצר ליד המילה נקודה.

השורות ובהן המילים Accept ו decline יעלמו ועליכם לחוץ על windows Offline installation, שם הקובץ שיישמר הוא j2re-1_4_2_12-windows-i586-p.exe.

5. לאחר ההורדה יש להפעיל את הקובץ על ידי לחיצה כפולה ולאשר בכל המסכים הדורשים זאת.

הורדה והתקנה של JFLAP

1. כותבי התוכנה מחלקים אותה בחינם ומבקשים מכם למלא שאלון פרטים קצר. השאלון נמצא בכתובת <http://cgi.cs.duke.edu/~rodger/jflap.cgi.html>. בסיום מילוי הפרטים לחצו על submit.

2. אם אין ברצונכם למלא פרטים אישיים, היכנסו לכתובת <http://www.jflap.org/jflaptmp> ותגיעו ישירות לדף הורדה. בדף זה גרסאות שונות של התוכנה. מדריך זה נכתב לגרסה היציבה ביותר והיא הגרסה: JFLAP.jar as of Feb 12 2005

3. לחצו על הקישור לקובץ זה ושמרו אותו במחשבכם.

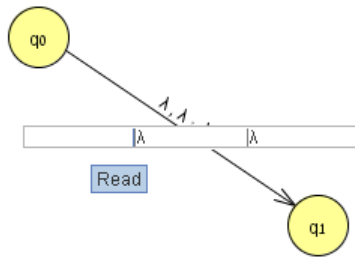
4. התוכנה מוכנה להפעלה – לחצו על הקובץ לחיצה כפולה והתכנה תיפתח.

הערה: ישנם מחשבים שדפדפן האינטרנט משנה את סיומת הקובץ באופן אוטומטי לסיומת zip. באם הדבר קרה במחשבכם עליכם פשוט לשנות סיומת זו בזמן שמירת הקובץ או לאחר ההורדה.

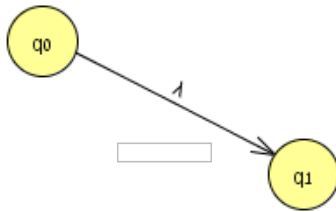
בעיה שהתגלו במהלך ההתנסות

במהלך התנסות ובדיקת דף הפעילות במחשבים שונים התגלתה בחלק מהמחשבים בעיה ביצירת מעבר ועריכתו. מקור הבעיה אינו ברור ובאם הפתרונות המוצגים בהמשך אינם יעילים או שהם מסורבלים מידי לביצוע ניתן פשוט לשמור את הקובץ, לסגור אותו ולפתוח אותו שוב.

תיאור הבעיה:



בצילום המסך משמאל ניתן לראות מצב עריכה ל אוטומט מחסנית. כדי להגיע למצב זה, יש לחוץ על לחצן הבחירה, לחוץ לחיצה בודדת או כפולה על התו, תיפתח חלונית ובה התו. כעת יש למחוק את התו להקליד את התו הרצוי וללחוץ עם העכבר על מקום כלשהו במסך. במקום הלחיצה על המסך ישנה אפשרות אחרת והיא לחוץ ENTER.



לעיתים החלונית לא נפתחת או שהיא נפתחת במקום מרוחק מאזור המעבר. ניסיון להתעלם מכך ופשוט להקליד את התו כמו גם הניסיון לחוץ עם העכבר על החלונית המרוחקת מוציא את המשתמש מכלי העריכה ומשאיר את התו המקורי.

(בצילום המסך הפעם מוצג אוטומט סופי אבל הבעיה זהה)

פתרון:

באוטומט סופי, הפתרון הפשוט הוא להשתמש רק בלחיצה בודדת(להימנע מלחיצה כפולה), להקליד את התו תוך התעלמות ממיקום החלונית וללחות במקום כלשהו במסך. באוטומט מחסנית יש לנקוט באותה שיטה ולעבור מתא לתא על ידי לחצן Tab במקלדת ולא על ידי העכבר.

כאמור אם אף אחת מהשיטות אינה עובדת יש לסגור ולפתוח את הקובץ שוב.

אינדקס פקודות

C

Change Label · 8
Clear · 8, 12
Clear All Labels · 8
Clear Label · 8
Compare Equivalence · 15
Complete · 15, 16
Convert · 15, 16, 17, 18
Convert FA to RE · 17, 18
Convert to DFA · 15, 16, 17

D

Dismiss Tab · 10, 12, 16
Display State Labels · 5, 8
Do It · 18
Done · 15, 16

E

Export · 18

F

FA to RE · 18
Fast Run · 10
File · 10, 12, 16
Final · 8

Finite Automatom · 4
Freeze · 9, 13

H

Highlight Nondeterminism · 17

I

I'm done · 10
Initial · 8
Input · 10

J

Java Runtime Environment · 4
JFLAP.jar · 4, 28

L

Layout Graph · 8

M

Multiple Run · 12, 22

P

Pushdown Automatom · 19

R

Rename States · 8
Reset · 11
Run Inputs · 12

S

Step · 11, 13, 14, 23
Step by State · 11, 13
Step with Closure · 11

T

Test · 15, 17
Thaw · 13
Trace · 11, 12, 22
Turing Machine · 24