

### מערכות הפעלה – תרגיל 3

בחלק זה יש לכתוב מחדש את התוכנית loopcmp (תרגיל 2) בצורה כזו שהיא תוכל להשתמש בתוכניות ההשוואה שהוכנו בתרגיל 1 מבלי שהתוכנית עצמה תכיל את הקוד לביצוע ההשוואות.

בניגוד למימוש בתרגיל 2 שבו התוכנית loopcmp תמכה, לאחר התוספת שלכם, בשלוש שיטות השוואה בלבד אשר מומשו כחלק מהתוכנית, המימוש הנוכחי לא יהיה מוגבל במספר שיטות ההשוואה וניתן יהיה להשתמש בתוכנית עם כל שיטת השוואה בה נחפוץ (ולא רק בשלוש אותן כבר הכנו) ללא צורך בהידור מחדש של התוכנית (cכפוף לכך שהמשתמש 'יספק' תוכנית השוואה מתאימה).

תיאור הקלט והפלט הנדרשים זהה למימוש loopcmp בתרגיל ההכנה (למעט ההגבלה ב-loopcmp לשלוש שיטות ההשוואה המסויימות בלבד שכעת כאמור הוסרה). אין להוסיף הדפסות 'עזר', הודעות שגיאה וכדומה.

תיאור התוכנית:

1. התוכנית תקבל כפרמטר בזמן הפעלת התוכנית את שיטת ההשוואה הרצויה.
2. לאחר מכן, כל עוד לא נגמר הקלט:
  - א. התוכנית קולטת שתי שורות של טקסט עד אשר הקלט הסטנדרטי מגיע ל-EOF (על מנת להזין EOF בקלט הסטנדרטי יש להקיש ctrl-d).
  - ב. לאחר קליטת שתי שורות (מחרוזות) התוכנית יוצרת תהליך בן על ידי קריאת המערכת fork.
  - ג. ככל שתהליך הבן נוצר בהצלחה:
    - i. על הבן לקרוא לקריאת המערכת execve (או לפונקציה execvp הקוראת לקריאת מערכת זו) ולהפעיל את תוכנית ההשוואה הרלוונטית.
    - ii. על האבא להמתין (קריאת המערכת wait) לסיום ההשוואה על ידי הבן ולהדפיס, באופן זהה להדפסה בתוכנית בתרגיל 2, את תוצאת ההשוואה (רצוי לחלץ את ערך ההחזרה של תהליך הבן על ידי שימוש במאקרו WEXITSTATUS המתואר בעמוד ה-man של קריאת המערכת wait).

ככלל, ערכי ההחזרה של התוכנית יהיו זהים לערכי ההחזרה כפי שמומשו בתרגיל 2, למעט התוספת שבמקרה של כישלון בקריאת מערכת על ערך ההחזרה להיות 2-.