

תכנות פונקציונלי בסביבת JVM

תיאור הצעה לפרויקט סיום

מטרת הפרויקט :

- מימוש עץ סייפות (suffix tree) עבור מחרוזת T בשפת Clojure.
- מתן מענה לשאלות מהצורה חיפוש תבנית P באורך m במחרוזת T.

תיאור הבעיה : בנייה יעילה של מבנה חדש שאינו ניתן לשינוי (immutable).

הגישה לפתרון :

לשם מימוש העץ נשתמש במבנים היעודיים (sequences / vectors / Hash map etc') של השפה שאינם ניתנים לשינוי (immutable) מרגע הזנת מחרוזת הקלט T ויצירת עץ הסייפות.

הודות לתכונת persistence בשפה, מובטח לנו ששלב בניית העץ יעשה בצורה יעילה על ידי שימוש בתת המבנה המשותף בין העץ בשלב ה-i לבנייתו והעץ בשלב ה-i+1 לבנייתו (ולא בצורה ליניארית שתכלול שכפול של כל צמתי העץ ה-i).

באשר לשאילתא- על מנת לתת מענה לשאילתא בזמן מיטבי נסרוק את מבנה הנתונים שיצרנו תוך שימוש ב Lazy evaluation; במקום לשמור את מבנה הנתונים כולו (שצפוי להיות גדול) בזיכרון, נבצע גישה אופטימלית לעץ לשם ביצוע השאילתא, תוך שימוש בספריות ה sequence ש-Clojure מציעה.

יתרון נוסף שמתקבל עבור פתרון בעיית עץ הסייפות בשפת Clojure, היא תכונת המקביליות (concurrency), כלומר מאחר שמרגע יצירתו המבנה לא עובר שינויים (read-only), וע"י כך הוא מתיר מספר גישות מקבילות לעץ בלא צורך בסינכרון ובאמצעי נעילה.