

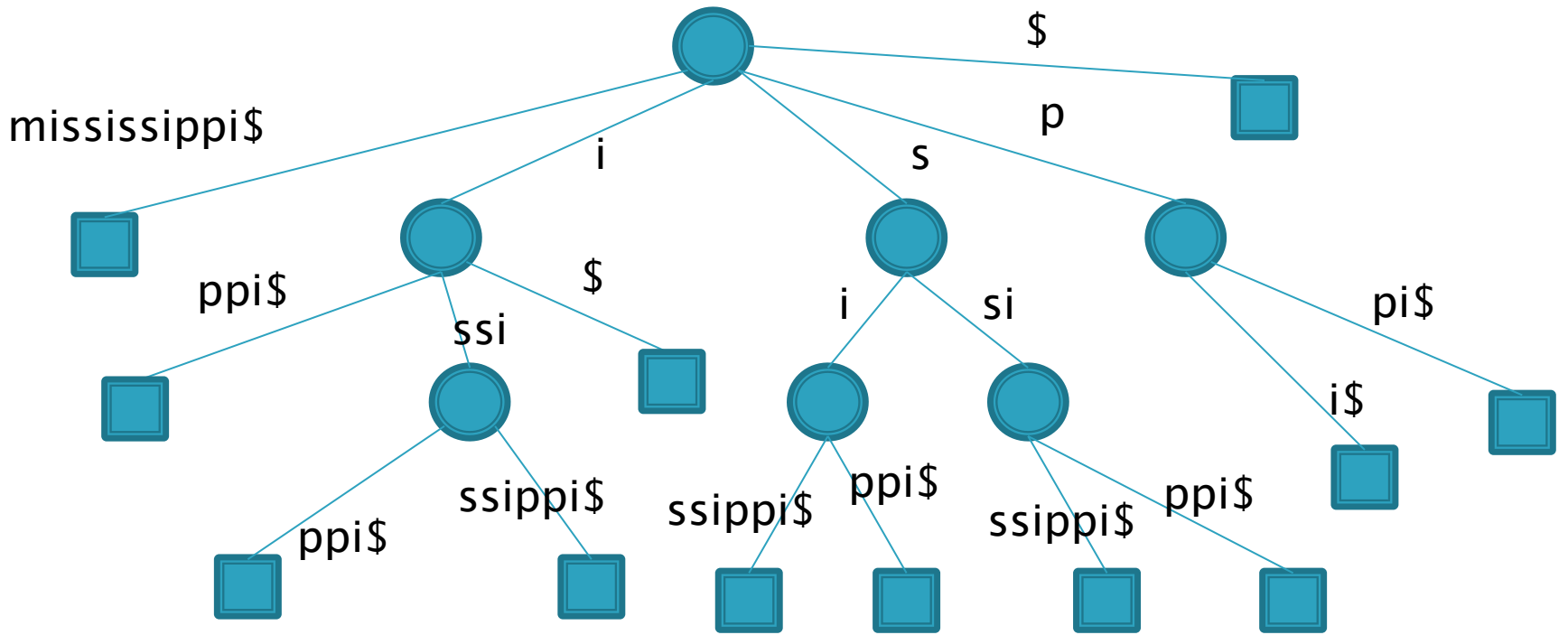
מבני נתונים

תרגיל 13, סמסטר א' תש"ע
עצי סייפות

בניית עץ סיפות

- ▶ נבנה עץ סיפות למילה mississippi
- ▶ לפני הבנייה:
 - נוסף \$ לסוף המחרוזת
 - נרשום את כל הסיפות

ליאור כבר בנו לנו את העץ



תרגיל 1

▶ תארו אלגוריתם הרץ בזמן לינארי, מקבל כקלט מחרוזת, ומוצא את תת-המחרוזת (לא ריקה) הקצרה ביותר שחוזרת בדיוק פעם אחת.

פתרון 1

- ▶ כל תת מחרוזת שחוזרת פעם אחת היא מסלול בעץ הסייפות שמסתיים בקשת שמובילה אל עלה.
- ▶ מכיוון שאנחנו מחפשים תת מחרוזת קצרה ביותר, ניקח מהקשת האחרונה במסלול רק את האות הראשונה.
- ▶ אם אות זאת היא \$, אז לא באמת מדובר בתת-מחרוזת שחוזרת פעם אחת במחרוזת המקורית.
- ▶ האלגוריתם:
 - בזמן לינארי נבנה עץ סייפות.
 - בזמן לינארי נמצא את המרחק באותיות אל כל קשת שמובילה לעלה.
 - נבחר את הקשת הקרובה ביותר מבניהן, שהתווית שלה היא לא רק \$.
 - תת המחרוזת המבוקשת היא המחרוזת שהובילה לקשת ועוד האות הראשונה שעליה.

תרגיל 2, סמטר ב' תשס"ז

▶ נגדיר תת מחרוזת z באורך k להיות חזרה מקסימלית ימנית במחרוזת S אם קיימים i ו- j , $i \neq j$ כך ש:

$$z = S_i S_{i+1} \dots S_{i+k-1}$$

$$z = S_j S_{j+1} \dots S_{j+k-1}$$

$$S_{i+k} \neq S_{j+k}$$

▶ תאר אלגוריתם ליניארי המקבל כקלט מחרוזת S ומספר k ומחשב את כל החזרות המקסימליות הימניות באורך k . עבור כל חזרה האלגוריתם מדווח את האינדקס בו מתחיל המופע הראשון שלה.

פתרון 2

- ▶ שוב, נבנה את עץ הסייפות
- ▶ הפעם נחפש צמתים פנימיים, במרחק k אותיות מהשורש

שאלה 3, סמסטר א' תשס"ח

▶ תארו אלגוריתם יעיל ככל האפשר, שמקבל מחרוזת בינארית s באורך n ומוצא את המחרוזת הארוכה ביותר p כך שגם p וגם המשלימה של p הן תתי מחרוזות של s .

פתרון 3

- ▶ בהינתן S , נחשב את S 'המשלימה של S .
- ▶ נבנה עץ סייפות מוכלל (generalized suffix tree) לשתי המחרוזות.
- ▶ נמצא את תת המחרוזת המשותפת הארוכה ביותר לפי האלגוריתם שראינו (נראה שבוע הבא) בכיתה.