



תכנית הלימודים אלגוריתמים – פרק בחירה בתכנית הלימודים במדעי המחשב לתיכון
31.3.2022

חברי הצוות:

ד"ר ראובן חוטובלי
פרופ' מיכל ארמוני

פריסת שעות

<u>פרק</u>	<u>הנושא</u>	<u>במעבדה</u>	<u>בכיתה</u>	<u>סה"כ שעות</u>
1	היכרות עם גרפים	1	3	4
2	ייצוג של גרפים	1	3	4
3	מסלולים קצרים ביותר ממקור יחיד	4	8	12
4	מסלולים קצרים ביותר בגרף עם משקלות אי-שליליים	3	7	10
5	מסלולים קצרים ביותר בגרף מכוון עם משקלות חיוביים ושליליים	2	4	6
6	סריקה לעומק	4	8	12
7	מיון טופולוגי ומסלולים קצרים בגמ"ל	2	4	6
8	עץ פורש מינימלי	3	6	9
9	זרימה ברשתות	4	8	12
10	קידוד ודחיסת נתונים	2	5	7
	בחנים ומבחנים	4	4	8
	<u>סה"כ שעות</u>	<u>30</u>	<u>60</u>	<u>90</u>

תוכן עניינים (הקלקה על מספר העמוד + לחצן Ctrl ביחד, תקפיץ לעמוד המתאים)

4	מבוא
4	קווים מנחים
5	פרק 1 : היכרות עם גרפים (4 שעות)
6	מושגים
6	מטרות ביצועיות
6	דגשים ודרכי הוראה
6	פרק 2 : ייצוג של גרפים (4 שעות)
7	מושגים
7	מטרות ביצועיות
8	דגשים ודרכי הוראה
9	פרק 3 : מסלולים קצרים ביותר ממקור יחיד (12 שעות)
9	מושגים
9	מטרות ביצועיות
10	דגשים ודרכי הוראה
10	אסטרטגיות אלגוריתמיות
11	פרק 4 : ממסלולים קצרים ביותר בגרף עם משקלות אי-שליליים (10 שעות)
11	מושגים
11	מטרות ביצועיות
12	דגשים ודרכי הוראה
13	אסטרטגיות אלגוריתמיות
13	פרק 5 : מסלולים קצרים ביותר בגרף מכוון עם משקלות חיוביים ושליליים (6 שעות)
13	מושגים
13	מטרות ביצועיות
14	דגשים ודרכי הוראה
14	אסטרטגיות אלגוריתמיות
15	פרק 6 : סריקה לעומק (12 שעות)
15	מושגים

15	מטרות ביצועיות
16	דגשים ודרכי הוראה
17	פרק 7 : מיון טופולוגי ומסלולים קצרים בגרף מכוון חסר מעגלים (6 שעות)
17	מושגים
17	מטרות ביצועיות
17	דגשים ודרכי הוראה
19	פרק 8 : עץ פורש מינימלי (9 שעות)
19	מושגים
19	מטרות ביצועיות
19	דגשים ודרכי הוראה
21	פרק 9 : זרימה ברשתות (12 שעות)
21	מושגים
21	מטרות ביצועיות
22	דגשים ודרכי הוראה
23	פרק 10 : קידוד ודחיסת נתונים (8 שעות)
23	מושגים
23	מטרות ביצועיות
23	דגשים ודרכי הוראה
24	פריסת שעות

מבוא

תכנית הלימודים במדעי המחשב נועדה לחשוף את התלמידים ליסודות הדיסציפלינה ולעקרונותיה (Gal-Ezer, Beer, Harel, and Yehudai, 1995), תוך שילוב מתמיד בהתנסות ויישום. התלמידים מכירים את התהליך של פתרון בעיות אלגוריתמיות, החל מניתוח בעיה, דרך מציאת פתרון אלגוריתמי עבורה ועד מימוש הפתרון בשפת תכנות.

שני פרקי החובה הראשונים ("יסודות מדעי המחשב") עורכים לתלמיד היכרות עם מושגים אלגוריתמים בסיסיים, פיתוח אלגוריתמים ויישומים ע"י תכנות. פרק החובה השלישי עוסק במושגים מתקדמים יותר תוך העמקה במבני נתונים. גם במהלכו פותרים התלמידים בעיות אלגוריתמיות ומיישמים אותם ע"י תכנות. הפתרונות הם מורכבים יותר ונעזרים במבני נתונים מסוגים שונים. גם בשלב פרקי החובה, נחשפים התלמידים לבעיות אלגוריתמיות יסודיות (כגון מיון וחיפוש) ואלגוריתמים קלאסיים עבורם.

פרק הבחירה "אלגוריתמים" מתמקד בחלקו הראשון של תהליך פתרון בעיות אלגוריתמיות – ניתוח בעיה נתונה ותכנון פתרון אלגוריתמי עבורה. לצורך כך התלמידים לומדים אלגוריתמים קלאסיים (מורכבים יותר מאשר אלו שפגשו בפרקי החובה), את עקרונותיהם והרעיונות הגלומים בהם. המיקוד של הפרק הוא בתכנון אלגוריתמים ואינו כולל יישום (תכנות).

הפרק הזה יחד עם פרק הבחירה המקביל לו ("מודלים חישוביים") מהווים את הפן התיאורטי בתכנית הלימודים, ועורכים לתלמידים היכרות עם העשייה התיאורטית במדעי המחשב ומאפייניה.

קווים מנחים

1. ההוראה הינה מבוססת בעיות (problem-based). כלומר, כל מושג ובפרט כל אלגוריתם, נלמד בהקשר, תוך כדי פתרון בעיה אלגוריתמית אליה הוא רלבנטי. הוראה זו משלבת שלוש גישות: (1) למידה מבוססת בעיות, (2) למידה ע"י עשייה (3) ולמידה בהקשר, כל אחת מהן ידועה כמקדמת למידה משמעותית. יתרה מזאת, הוראה זו משקפת עיסוק אותנטי במדעי המחשב המטפל בפתרון בעיות אלגוריתמיות.
2. כל פרק פותח בבעיה בסיסית, אך ניתן כמובן להשתמש בבעיות מתאימות אחרות, לפי בחירתם וכיד דמיונם של המורים.
3. התלמידים לא רק משתמשים באלגוריתמים נתונים, אלא גם מפתחים אלגוריתמים על בסיס האלגוריתמים הנלמדים. לכן, בכל פרק חשוב לתת בעיות נוספות שלצורך פתרון ייעשה שימוש באלגוריתמים שנלמדו בו – כלשונם (כלומר, כקופסאות שחורות) או תוך הכנסת שינויים והתאמות באלגוריתמים שנלמדו.

4. חלקים בלתי נפרדים מפתרון בעיה אלגוריתמית הם הוכחת נכונות הפתרון וניתוח הסיבוכיות. התלמידים ינתחו את סיבוכיות הפתרונות שלהם אך לא תמיד יידרשו לתת הוכחה מלאה לנכונותם משום שבמקרים רבים ההוכחות מורכבות ומחייבות כלים שאינם מוכרים לתלמידים (כמו אינדוקציה מבנית). יחד עם זאת, כדאי לנצל הזדמנויות בהן הוכחת הנכונות מתאימה לרמת הקושי של התכנית כדי להדגים הוכחת נכונות מלאה.
5. בנוסף להיכרות עם אלגוריתמים חשובים התלמידים יכירו אסטרטגיות אלגוריתמיות שונות (רדוקציה, אסטרטגיה חמדנית, תכנון דינמי). גם אסטרטגיות אלו ילמדו מתוך הקשר, ותמיד לצורך פתרון בעיה עבורה הן רלוונטיות.
6. החוט המקשר בין האלגוריתמים המוצגים בקורס הוא נתונים – ייצוג וקידוד. התלמידים יעסקו בחלקו הראשון של הפרק בבעיות שניתן לייצג אותן ואת מרחב הנתונים שלהן ע"י גרפים ולאחר מכן יעסקו באלגוריתם לקידוד נתונים.
7. במסגרת התוכנית רעיונות חוזרים ומופיעים בהקשרים שונים ובצורות שונות. חשוב מאוד ליצור קשר מפורש בין ההופעות השונות של אותו רעיון. בכל פעם שהוא מופיע שוב להתייחס לכך במפורש ולהזכיר את משמעותו. למשל – הרעיון של הפשטה (אבסטרקציה) של נתונים או הרעיון של אסטרטגיה אלגוריתמית לפתרון בעיות. התייחסות מפורשת לרעיון על מופעיו השונים תאפשר הפנמה משמעותית של הרעיון תוך הכללה שלו. כפועל יוצא יורחב השימוש בו גם בהקשרים שונים מאלו המפורשים בחומר הלימוד.
8. כדאי לשלב בעיות מעניינות ורלוונטיות לתלמידים, מתחומים שונים.

פרק 1 : היכרות עם גרפים (4 שעות)

בעיה בסיסית: את מבנה הנתונים המופשט של גרף ומושגים בסיסיים הקשורים אליו ניתן להציג בעזרת בעית "המסיבה": אנשים מגיעים למסיבה ולוחצים זה את ידו של זה. יש לחשב את מספר לחיצות הידיים בתנאים שונים של הבעיה.

הבעיה הבסיסית הזאת מאפשרת להגדיר את המושגים הבסיסיים של צמתים (אנשים), קשתות (לחיצות ידיים) ודרגה של צומת (מספר הידיים שלחץ אדם). בעזרת וריאציות על הבעיה הראשונית אפשר להגדיר מושגים נוספים (גרף מכוון ולא מכוון, גרף שלם כמו גם גרף לא שלם).

בעיה בסיסית נוספת: בעיית הגשרים של אוילר (Euler), וכמובן הווריאציה שלה בציור שניתן לצייר ביד אחת בלי להרים את העט מהדף. בעיה זאת מסייעת להדגים את המושגים של מסלול, מסלול פשוט, מסלול מעגלי ומסלול מעגלי פשוט.

מושגים

- גרף
- צמתים
- קשתות
- גרף מכוון וגרף לא מכוון
- דרגה של צומת
- גרף מלא
- מסלול ומסלול מעגלי

מטרות ביצועיות

1. התלמיד יסביר מהו גרף מכוון, לא מכוון ומושקל.
2. התלמיד יסביר : דרגת כניסה, דרגת יציאה, דרגה, מסלול ומעגל.
3. התלמיד יסביר המושגים : בעבור גרף לא מכוון - רכיב קשיר וגרף קשיר ובעבור גרף מכוון - מהו רק"ח, גרף קשיר היטב.
4. התלמיד יסביר מהו גרף שלם, גרף דו צדדי ותת גרף.
5. התלמיד יסביר הקשר בין סכום הדרגות של קודקודי הגרף לבין מספר הקשתות בגרף.
6. התלמיד יסביר מהם התנאים על מנת שגרף יהיה עץ.

דגשים ודרכי הוראה

- הבעיה הראשונה ("המסיבה") היא בעיית חישוב מתמטית ולא בעיה אלגוריתמית, אבל היא נותנת הקשר להיכרות עם המושגים הבסיסיים של גרף.
- הבעיה השנייה (הגשרים של אוילר) היא בעיה אלגוריתמית. מאחר שאינה פשוטה מתאים לפתור אותה במליאה, כלומר, תוך כדי דיון שבו בשלב ראשון תתפתח הבנה של הבעיה ורק אז יפותח אלגוריתם (בכתיבה כללית).

פרק 2: ייצוג של גרפים (4 שעות)

בעיה בסיסית: בעיית הגשרים של אוילר שהוצגה בפרק הקודם.

תוך התמודדות עם הבעיה של מציאת מסלול אוילר ניתן להבהיר את המשמעות של גרף כמבנה נתונים מופשט ופעולות נדרשות עליו (למשל, מהי רשימת הקשתות של צומת מסוים? מה הדרגה של צומת? יש לדון בדרכים לייצוג גרף (עבור בעיית מסלול אוילר עולה באופן טבעי המבנה של רשימת סמיכויות) וייצוג מסלול, ובהתאם לכך לנתח את סיבוכיות של הפתרון.

בעיה בסיסית נוספת: מבט חוזר על בעיות לחיצות הידיים (שהוצגה בפרק הקודם) יכול להוביל למבנה נתונים של מטריצת סמיכויות.

מושגים

- רשימת סמיכויות
- מטריצת סמיכויות

מטרות ביצועיות

1. התלמיד יסביר את הפעולות הבסיסיות על גרף (הוספת קשת, הוספת קשת משוקללת, הסרת קשת, הסרת קשת משוקללת, בדיקת קיום של קשת, יצירת גרף ריק, הוספת קודקוד, הסרת קודקוד).
2. התלמיד יסביר את הייצוג של גרף באמצעות מטריצה ריבועית.
3. התלמיד יסביר את הייצוג של גרף באמצעות רשימות שכנות.
4. התלמיד יפתח פתרון אלגוריתמי למימוש הפעולות בסעיף (1), כאשר גרף מיוצג באמצעות מטריצה ריבועית, ויחשב את סיבוכיות זמן הריצה שלהן.
5. התלמיד יפתח פתרון אלגוריתמי למימוש הפעולות בסעיף (1), כאשר גרף מיוצג באמצעות רשימות שכנות ויחשב את סיבוכיות זמן הריצה שלהן.
6. התלמיד יפתח, בגישה מודולרית, פתרונות אלגוריתמים המשתמשים בייצוגים שונים של גרף.
7. התלמיד יפתח פתרונות אלגוריתמיים בעבור הבעיות: חישוב דרגת כניסה, דרגת יציאה, דרגה של קדקודי הגרף, כאשר משתמשים בייצוגים שונים של גרף ויחשב את סיבוכיות זמן הריצה שלהן.
8. התלמיד יסביר מהו גרף הפוך ויפתח פתרון אלגוריתמי ליצירת גרף הפוך כאשר משתמשים בייצוגים שונים של גרף ובנוסף יחשב את סיבוכיות זמן הריצה שלהן.
9. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים שימוש בייצוגים שונים של גרף.

דגשים ודרכי הוראה

- הנושא מאפשר לחדד את העקרונות של מבנה נתונים מופשט. הפעולות הנדרשות הן זהות אך בכל ייצוג הן מקבלות מימוש בצורה אחרת וסיבוכיות שונה.
- הנושא מאפשר לחזור ולחדד את רמות ההפשטה של ייצוג נתונים. גרף הוא מבנה נתונים מופשט שיכול בתורו להיות ממומש על ידי רשימה מקושרת של רשימות מקושרות, כלומר דרך מבנה נתונים מופשט אחר.
- מפרק זה ואילך עבור כל אלגוריתם שילמד יש להתייחס לניתוח הסיבוכיות שלו, גם אם זה לא נאמר במפורש בתכנית.

פרק 3: מסלולים קצרים ביותר ממקור יחיד (12 שעות)

בעיה בסיסית: חקלאי בעל מספר מטעי תפוחים הפזורים בשטחים סביבו ודרכים חקלאיות המחברות בין המטעים. החקלאי רוצה לסמן לעצמו עבור כל אחד מהמטעים שברשותו באיזו דרך יוכל להגיע אל המטע כך שיעבור דרך כמה שפחות מטעים בדרכו אל היעד. פתרון לבעיה הוא האלגוריתם של חיפוש לרוחב (BFS).

האסטרטגיה האלגוריתמית של רדוקציה תעלה באופן טבעי בפרק ע"י הצגת בעיות שאפשר לפתור אותן ע"י שימוש במציאת מסלולים קצרים ממקור יחיד כקופסה שחורה. למשל – אצל חקלאי אחר יש הבדלים משמעותיים במרחקים בין המטעים. חלק מהמטעים קרובים יחסית זה לזה, כחצי ק"מ, אך יש מטעים שהמרחק ביניהם כפול מזה, כק"מ. כיצד ימצא החקלאי עבור כל מטע מהי הדרך הקצרה ביותר המגיעה אל היעד?

מושגים

- אורך מסלול
- מסלול קצר ביותר
- BFS
- עץ ועץ פורש של גרף
- קשירות של גרף

מטרות ביצועיות

1. התלמיד יסביר מהו אורך מסלול ומהו מסלול קצר ביותר.
2. התלמיד יסביר את אופן מציאת המסלול הקצר ביותר באמצעות אלגוריתם BFS.
3. התלמיד יסביר את סיבוכיות זמן הריצה של BFS כאשר גרף מיוצג באמצעות רשימות שכנות.
4. התלמיד יסביר מהו עץ פורש BFS.
5. התלמיד יסביר כיצד לקבוע אם בגרף נתון קיים מעגל.
6. התלמיד יסביר כיצד לקבוע אם גרף נתון לא מכון הוא גרף קשיר או לא קשיר.
7. התלמיד יתכנן אלגוריתם הפותר בעיה נתונה תוך שימוש ב-BFS כקופסה שחורה.
8. התלמיד ישתמש בטיפוסי נתונים מופשטים (מחסנית/ תור/ רשימה) הנלמדים במבנה נתונים לצורך פתרון בעיות הקשורות לסריקה לרוחב בגרפים.
9. התלמיד יסביר מדוע אלגוריתם מסוים הוא נכון.
10. התלמיד יסביר מהי רדוקציה.

11. התלמיד יפתור בעיות תוך שימוש ברדוקציה ושימוש ב-BFS למציאת מסלולים קצרים ממקור יחיד כקופסה שחורה.

דגשים ודרכי הוראה

- המקרה הפשוט של צומת יעד מסוים יחיד.
- מימוש BFS בעזרת תור (כמבנה נתונים מופשט). זאת דוגמה נוספת למבנה בתוך מבנה.
- המשמעות של BFS כאלגוריתם לסריקת גרף תוך קישור לבעיות סריקה שכבר הכירו בפרקים הקודמים בתכנית הלימודים – סריקת רשימה וסריקת עץ. מוצע לדון במטרות של סריקה – לצורך חיפוש איבר, לצורך חישוב מרחקים. להדגיש את אפיון הסריקה כסריקה לרוחב.
- אלגוריתם הסריקה יוצר עץ פורש מכוון (קישור למבנה הנתונים של עץ שהכירו ביחידה העוסקת במבני נתונים).
- BFS בגרף מכוון – את המקרה המכוון ניתן להציג דרך בעיה דומה שבה יש דרכים דו-סטטריות וחד-סטטריות. גם זה פתרון ע"י רדוקציה – רדוקציה לבעיה הבסיסית של BFS בגרף לא מכוון.
- וריאציה על הבעיה המקורית יכולה להוביל להגדרת קשירות ושימוש ב-BFS לבדוק קשירות של גרף. למשל: בעונת הגשמים ייתכן שחלק מהדרכים המקשרות בין מטעים חסומות עקב הצפה ויש למצוא כעת את הדרך הקצרה ביותר לכל מטע. האם עדיין אפשר להגיע לכל מטע?
- בנוסף לחשיבות של חשיפה ראשונה לרדוקציה כאל אסטרטגיה אלגוריתמית, אלגוריתמים שמבוססים על רדוקציה מאפשרים להתנסות בהוכחת נכונות של אלגוריתמים כי אלו הוכחות פשוטות יחסית שמניחות את נכונות האלגוריתם הבסיסי.

אסטרטגיות אלגוריתמיות

- רדוקציה

פרק 4: ממסלולים קצרים ביותר בגרף עם משקלות אי-שליליים (10 שעות)

בעיה בסיסית: נהג משווק סחורה לכמה חנויות צעצועים מפוזרות בעיר. ישנם כבישים באורכים שונים המחברים בין חנויות. הנהג רוצה למצוא עבור כל חנות מהי הדרך הקצרה להגיע אליה.

פתרון לבעיה הוא האלגוריתם של דייקסטרה (Dijkstra).

זאת הרחבה טבעית של הבעיה שהייתה במרכז הפרק הקודם. במקום מטעים מפוזרים על שטח קטן יחסית יש חנויות צעצועים במרחקים השונים משמעותית זה מזה. כעת יש חשיבות למרחקים בין החנויות. פחות חשוב דרך כמה חנויות הנהג עובר בדרכן לכל חנות, וחשוב יותר מה אורך הדרך שעליו לנסוע עד הגיעו לכל חנות.

האלגוריתם של דייקסטרה נשען על שתי אסטרטגיות אלגוריתמיות ומאפשר היכרות ראשונה עם שתיהן – הוא חמדני (בכל שלב נבחר הצומת שקרוב יותר לצומת המקור) ובנוסף, המסלול הקצר ביותר לכל צומת נבחר ע"י פתרון של תתי-בעיות, חישוב תתי-מסלולים בדרך ושימוש בתתי המסלולים הקצרים יותר, כלומר יש כאן שימוש באסטרטגיה של תכנות דינמי.

מושגים

- משקולות לקשתות
- האלגוריתם של דייקסטרה

מטרות ביצועיות

1. התלמיד יסביר מהו משקל מסלול ומהו מסלול קצר ביותר במשקל.
2. התלמיד יסביר את אופן מציאת המסלול הקצר ביותר במשקל באמצעות האלגוריתם של דייקסטרה (Dijkstra).
3. התלמיד יסביר את סיבוכיות זמן הריצה של אלגוריתם דייקסטרה (Dijkstra), כאשר גרף מיוצג באמצעות מטריצת שכנות.
4. התלמיד יסביר מהו עץ מסלולים קצרים במשקל.
5. התלמיד יסביר באיזה מקרה לא ניתן להשתמש באלגוריתם של דייקסטרה.
6. התלמיד יסביר מהו תור עדיפויות, באיזה אופן משתמש האלגוריתם של דייקסטרה בתור עדיפויות ולמה הכרחי שזה יהיה תור עדיפויות ולא תור רגיל.
7. התלמיד יסביר מהי גישה חמדנית ואיך הגישה הזו באה לידי ביטוי באלגוריתם של דייקסטרה.
8. התלמיד יסביר מהי אסטרטגיית תכנות דינמי ואיך האסטרטגיה הזו באה לידי ביטוי באלגוריתם של דייקסטרה.
9. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש באלגוריתם של דייקסטרה כקופסה שחורה.

10. התלמיד יסביר מדוע אלגוריתם מסוים הוא נכון.

11. התלמיד יפתור בעיות תוך שימוש ברדוקציה ושימוש באלגוריתם של דייקסטר כקופסה שחורה.

דגשים ודרכי הוראה

- הבעיה בפרק היא דוגמה שמראה היטב כיצד שינוי קל בתנאי בעיה (אורכים לקשתות) יכול להקשות אותה משמעותית. התנסות בכיתה יכולה להוביל למסקנה ש-BFS כבר אינו מתאים ויש צורך באלגוריתם אחר.
- בפרק הקודם הודגם שימוש במבנה נתונים מופשט פנימי (תור) בתוך האלגוריתם. האלגוריתם של דייקסטר משתמש גם הוא במבנה נתונים מופשט פנימי, אך מורכב יותר – תור עדיפויות. יש לדון במה תור עדיפויות דומה לתור רגיל ובמה הוא שונה ממנו? באיזה אופן משתמש האלגוריתם של דייקסטר בתור עדיפויות ולמה הכרחי שזה יהיה תור עדיפויות ולא תור רגיל?
- בניתוח הסיבוכיות של האלגוריתם של דייקסטר יש חשיבות רבה לתור עדיפויות ולסיבוכיות הפעולות הנעשות עליו. יש לאתגר: כיצד תלויה סיבוכיות האלגוריתם בסיבוכיות הפעולות השונות? אפשר לדון בדרכים שונות למימוש תור עדיפויות (למשל, בעזרת רשימה מקושרת) ולסיבוכיות המתקבלת בעזרתן. אבל, מאחר שהדרכים היעילות יותר למימוש תור עדיפויות (ערימה בינארית וערימת פיבונאצ'י) הן מעבר לחומר הלימוד, זאת הזדמנות טובה לחזור ולדון במבנה נתונים מופשט כקופסה שחורה. לצורך ניתוח סיבוכיות האלגוריתם של דייקסטר מספיק לדעת שקיימת דרך יעילה לממש תור עדיפויות כך שהפעולות שנדרשות באלגוריתם של דייקסטר יעבדו בסיבוכיות שמוצהרת ע"י הממשק של המבנה.
- מגבלות האלגוריתם: בדומה למעבר מגרף חסר משקלות לגרף עם משקלות אי-שליליים, שינוי נוסף בתנאי הבעיה (אפשרות למשקלות שליליים) ייצור בעיה קשה יותר שלא נפתרת ע"י האלגוריתם של דייקסטר.
- למעשה, האסטרטגיה החמדנית היא זאת שמכשילה את האלגוריתם של דייקסטר כאשר ייתכנו משקלות שליליים, והדגמה של זה מסייעת להבהיר את אופיה של האסטרטגיה החמדנית שאינה מסתכלת קדימה ומתיימרת מוגבלת כזאת אינה מתאימה.
- דיון כללי באסטרטגיות אלגוריתמיות – בשלב הזה התלמידים הכירו כבר שלוש אסטרטגיות אלגוריתמיות חדשות. זה הזמן למבט רפלקטיבי אחורה, על יחידות קודמות. האם הכירו אסטרטגיות נוספות בפרקים קודמים של תכנית הלימודים, גם אם לא קראו להן כך במפורש? זה המקום להזכיר את האסטרטגיה של "הפרד ומשול" שבאה לידי ביטוי בחיפוש בינארי ובמיון מיזוג.

אסטרטגיות אלגוריתמיות

- אלגוריתם חמדני
- תכנות דינמי

פרק 5: מסלולים קצרים ביותר בגרף מכוון עם משקלות חיוביים ושלייליים (6 שעות)

מושגים

- משקלות שלייליים לקשתות
- מעגל שליילי והאלגוריתם של בלמן פורד

מטרות ביצועיות

1. התלמיד יסביר מהו משקל שליילי ומהו מסלול מעגלי במשקל שליילי.
2. התלמיד יסביר את אופן מציאת המסלול הקצר ביותר במשקל באמצעות האלגוריתם של בלמן-פורד (Belman - Ford).
3. התלמיד יסביר את סיבוכיות זמן הריצה של אלגוריתם בלמן-פורד כאשר גרף מיוצג באמצעות רשימות שכנות.
4. התלמיד יסביר מהו עץ מסלולים קצרים במשקל בגרף מכוון.
5. התלמיד יסביר באיזה מקרה לא ניתן להשתמש באלגוריתם של בלמן-פורד.
6. התלמיד יסביר מהי אסטרטגיית תכנות דינמי ואיך האסטרטגיה הזו באה לידי ביטוי באלגוריתם של בלמן-פורד.
7. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש באלגוריתם של בלמן-פורד כקופסה שחורה.
8. התלמיד יסביר מדוע אלגוריתם מסוים הוא נכון.
9. התלמיד יפתור בעיות תוך שימוש ברדוקציה ושימוש באלגוריתם של בלמן-פורד כקופסה שחורה.

דגשים ודרכי הוראה

- גם בפרק זה נראה כיצד שינוי קל בתנאי בעיה (אורכים שליליים לקשתות) יכול להקשות אותה משמעותית. התנסות בכיתה יכולה להוביל למסקנה שדייקסטר-כבר אינו מתאים ויש צורך באלגוריתם אחר.
- מהי המשמעות של מעגל באורך שלילי ומדוע מעגל שלילי מכשיל את האלגוריתם של בלמן פורד.
- כמו בפרק הקודם, דיון כללי באסטרטגיות אלגוריתמיות

אסטרטגיות אלגוריתמיות

- תכנות דינמי

פרק 6: סריקה לעומק (12 שעות)

בעיה בסיסית: יציאה ממבוך. את המבוך אפשר לייצג ע"י גרף – הצטלבויות ושבילים שעוברים ביניהן. יש הצטלבויות שיוצאים מהן שבילים רבים, ויש שבילים ללא מוצא, שאם מגיעים לקצותיהם צריך להסתובב אחורה ולחזור להצטלבות האחרונה שיצאנו ממנה. הליכה לא זהירה עלולה להוביל לסיבוב חסר תוחלת במעגל ולאי-יציאה מהמבוך.

הפתרון – עיקרון הימנית. התרגום של העיקרון לאלגוריתם חד-משמעי ולייצוג של מבוך ע"י גרף מוביל לאלגוריתם של חיפוש לעומק (DFS).

מושגים

• DFS

מטרות ביצועיות

1. התלמיד יסביר מהי סריקה לעומק תוך שימוש באלגוריתם של DFS.
2. התלמיד יסביר את סיבוכיות זמן הריצה של האלגוריתם DFS כאשר גרף מיוצג באמצעות מטריצת שכנות.
3. התלמיד יסביר מהו עץ פורש DFS.
4. התלמיד יסביר כיצד אפשר לאפיין את קשתות הגרף (מכוון/ לא מכוון) ביחס לקשתות העץ הפורש.
5. התלמיד יסביר כיצד לקבוע אם בגרף נתון קיים מעגל.
6. התלמיד יסביר כיצד לקבוע אם גרף נתון לא מכוון קשיר או לא קשיר.
7. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש ב-DFS כקופסה שחורה.
8. התלמיד ישתמש בטיפוסי נתונים מופשטים (מחסנית / תור / רשימה) הנלמדים במבנה נתונים לצורך פתרון בעיות הקשורות לסריקה לעומק בגרפים.
9. התלמיד יסביר מדוע אלגוריתם מסוים הוא נכון.
10. התלמיד יפתור בעיות תוך שימוש ברדוקציה ושימוש ב-DFS כקופסה שחורה.
11. התלמיד יסביר כיצד ניתן למצוא רכיבי קשירות חזקה (רק"חים) תוך שימוש ב-DFS כקופסה שחורה.
12. התלמיד יסביר מהו גרף על וכיצד ניתן למצוא גרף על תוך שימוש ברכיבי קשירות חזקה (רק"חים).

דגשים ודרכי הוראה

- מוצע לקיים באן דיון בכך שאפשר למצוא פתרון (עיקרון היד הימנית) בעולם הבעיה, לפני שנעשה מידול לעולם של מדעי המחשב. המידול לבעיה אלגוריתמית בתורת הגרפים מאפשר להגיע לניסוח מדויק של הפתרון כולל ניתוח סיבוכיות.
- סריקה לעומק (DFS) מול סריקה לרוחב (BFS).
- סריקה לעומק בגרף לא קשיר.
- מדוע רקורסיה היא טבעית עבור סריקה לעומק?
- העץ הפורש שנוצר בסריקה לעומק, בהשוואה לעץ הפורש שנוצר בסריקה לרוחב. כיצד אפשר לאפיין את קשתות הגרף (מכוון/ לא מכוון) ביחס לקשתות העץ הפורש?
- כיצד אפשר למצוא רכיבי קשירות חזקה (רק"חים) תוך שימוש ב- DFS ובגרף הפוך ומהם למצוא את גרף על.

פרק 7: מיון טופולוגי ומסלולים קצרים בגרף מכוון חסר מעגלים (6 שעות)

בעיה בסיסית: ההורים נסעו לשבוע, וכל מטלות הבית הן באחריותכם. ויש הרבה... אתם מכינים רשימה של כל המטלות שעליכם לבצע. למשל – להשקות את העציצים, לעשות קניות, לטייל עם הכלב, לבשל, לכבס, לשאוב אבק, לקפל כביסה, לשטוף את הרצפה, לאסוף את כל הבגדים המלוכלכים שזרוקים על הרצפה בחדר שלכם, ועוד כמה וכמה.

אחרי שהכנתם את הרשימה אתם רוצים לארגן סדר משימות שתפעלו לפיו. יש מטלות שהסדר ביניהן לא חשוב. למשל, לא משנה אם קודם תכבסו או קודם תשטפו את הרצפה. אבל במקרים רבים הסדר חשוב מאוד. למשל – את הבגדים המלוכלכים צריך לאסוף מהרצפה לפני ששוטפים אותה וגם לפני שמכבסים. קודם צריך לכבס ורק אחר כך לקפל כביסה. קודם צריך לעשות קניות ורק אחר כך לבשל. קביעת סדר המטלות הכללי צריכה להתחשב באילוצי הסדר האלו.

זאת בעיית מיון טופולוגי ואפשר לפתור אותה ע"י אלגוריתם למיון טופולוגי שמבוסס על DFS.

מושגים

- מיון טופולוגי
- מסלולים קצרים בגרף מכוון חסר מעגלים

מטרות ביצועיות

1. התלמיד יסביר מהו מיון טופולוגי וכיצד ניתן להשיג מיון כזה תוך שימוש באלגוריתם DFS.
2. התלמיד ישתמש בטיפוסי נתונים מופשטים (מחסנית / תור / רשימה) הנלמדים במבנה נתונים לצורך מיון טופולוגי.
3. התלמיד יסביר את סיבוכיות זמן הריצה של אלגוריתם מיון טופולוגי כאשר גרף מיוצג באמצעות מטריצת שכנות.
4. התלמיד יסביר כיצד ניתן לקבוע אם בגרף נתון קיים מעגל תוך שימוש במיון טופולוגי.
5. התלמיד יסביר על אילו גרפים ניתן להפעיל את האלגוריתם מיון טופולוגי.
6. התלמיד יסביר כיצד למצוא מסלול קצר ביותר בגרף מכוון חסר מעגלים. (DAG)
7. התלמיד יתכנן אלגוריתם הפותר בעיה נתונה תוך שימוש במיון טופולוגי כקופסה שחורה.
8. התלמיד יסביר אלגוריתם נתון המשתמש במיון טופולוגי כקופסה שחורה.

דגשים ודרכי הוראה

- האם תמיד ניתן לבצע מיון טופולוגי? מה קורה בגרף שמכיל מעגל?

- הישענות על אלגוריתם קיים (DFS לגרף מכוון) כדי לפתור בעיה שונה.
- הישענות על אלגוריתם מיון טופולוגי כדי לפתור בעיית מסלול קצר ביותר בגרף מכוון חסר מעגלים.

פרק 8: עץ פורש מינימלי (9 שעות)

בעיה בסיסית: נחזור מעט בהיסטוריה לשלב שבו התרבות המערבית עברה מתחבורה שאינה ממונעת לתחבורה ממונעת. יש ערים ודרכים המחברות ביניהן ומתאימות לכרכרות סוסים. אבל עם המצאת המכוניות והתרחבות השימוש בהן, יש לסלול חלק מהדרכים כך שיתאימו לנסיעה ברכב. הדרישה הבסיסית היא שיהיה ניתן להגיע לכל עיר ועיר, אך מאחר שהחומר והעבודה הנדרשים לסלילת כבישים הם יקרים המטרה היא לסלול כמה שפחות קילומטרים.

הבעיה תיפתר ע"י האלגוריתם של קרוסקל (Kruskal).

מושגים

- עץ פורש מינימלי

מטרות ביצועיות

1. התלמיד יסביר מהו עץ פורש מינימלי.
2. התלמיד יסביר את האסטרטגיה של האלגוריתם של קרוסקל (Kruskal).
3. התלמיד יסביר את סיבוכיות זמן הריצה של האלגוריתם של קרוסקל.
4. בניתוח סיבוכיות של האלגוריתם של קרוסקל יש חשיבות שהתלמיד ידע להסביר כיצד תלויה סיבוכיות האלגוריתם בסיבוכיות הפעולות השונות.
5. מאחר שהדרכים היעילות יותר למימוש הפעולות האלו הן מעבר לחומר הלימוד התלמיד יסביר שניתן להשיג סיבוכיות הטובה ביותר בעבור האלגוריתם קרוסקל תוך שימוש במבנה נתונים מופשט כקופסה שחורה.
6. התלמיד יסביר מהו ההבדל בין עץ פורש מינימלי לבין עץ המסלולים הקצרים שהתקבלו לפי BFS או לפי דייקסטרה.
7. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש באלגוריתם של קרוסקל כקופסה שחורה.
8. התלמיד יפתור בעיות תוך שימוש ברדוקציה ושימוש באלגוריתם של קרוסקל כקופסה שחורה.
9. התלמיד יסביר איך הגישה חמדנית באה לידי ביטוי באלגוריתם של קרוסקל ומדוע גישה זו מניבה פתרון אופטימלי למציאת עץ פורש מינימלי.

דגשים ודרכי הוראה

- נדון בהבדל בין הדרישות בבעיות של מסלולים קצרים ביותר לדרישות הבעיה של עץ פורש מינימלי. האם העץ הפורש שמיוצר ע"י BFS, בגרף שבו לכל הקשתות משקל זהה, הוא בהכרח עץ פורש מינימלי?

- מה ההבדל בין מציאת עץ פורש מינימלי למציאת מסלולים קצרים ביותר בגרף עם משקלות אי-שליליים?
- הפתרון מדגים שוב שימוש באסטרטגיה חמדנית.

פרק 9: זרימה ברשתות (12 שעות)

בעיה בסיסית: נתאר לעצמנו חומר העובר דרך מערכת, ממקור, שם מיוצר החומר, לבור, שם הוא נצרך. המקור מייצר את החומר בקצב קבוע כלשהו והבור צורך את החומר באותו קצב. זרימת החומר בכל נקודה במערכת היא הקצב שבו החומר נע.

על כל קשת מכוונת ברשת זרימה ניתן לחשוב כעל מוליך של חומר. כל מוליך הוא בעל קיבול ידוע, הנתון על ידי הקצב המקסימלי שבו יכול חומר לזרום דרך המוליך.

קודקודים הם נקודות הצטלבות של המוליכים, וחוף מאשר במקור או בבור, החומר זורם דרכם אך אינו נאסף בהם. במילים אחרות, הקצב שבו חומר נכנס לקודקוד חייב להיות שווה לקצב שבו החומר יוצא מהקודקוד. תכונה זו נקראת "שימור הזרימה".

כאשר החומר הוא זרם חשמלי, זהו חוק הזרמים של קירכהוף (Kirchhoff's Current Law).

בעיית הזרימה המקסימלית היא הבעיה הפשוטה ביותר הנוגעת לרשתות זרימה. היא שואלת: "מהו הקצב הגבוה ביותר שבו ניתן להעביר חומר מן המקור אל הבור מבלי להפר שום אילוץ קיבול?".

בעיה בסיסית יכולה להיות גם הפצת סחורה או בתחום התקשורת וכדומה.

הבעיה תיפתר ע"י אלגוריתם פורד-פולקרסון.

מושגים

- רשת זרימה
- קיבול
- זרם
- מקור
- בור
- קשת שיורית ורשת שיורית
- מסלול שיפור
- חתך וחתך מינימלי
- זרם מקסימלי
- גרף דו צדדי
- זיווג וזיווג מקסימלי

מטרות ביצועיות

1. התלמיד יסביר מהי רשת זרימה, מהו בור.
2. התלמיד יסביר מהו קיבול, מהו זרם וערך הזרם.
3. התלמיד יסביר מהי קשת שיורית ורשת שיורית.

4. התלמיד יסביר מהו מסלול שיפור.
5. התלמיד יסביר את האסטרטגיה של האלגוריתם של פורד-פולקרסון הבסיסי.
6. התלמיד יסביר את סיבוכיות זמן הריצה של האלגוריתם של הופמן.
7. התלמיד יסביר מהו חתך ומהו חתך מינימלי.
8. התלמיד יסביר הקשר בין חתך מינימלי לבין זרם מקסימלי.
9. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש באלגוריתם של פורד פולקרסון כקופסה שחורה.
10. התלמיד יסביר מהי רשת עם מקורות ובורות מרובים.
11. התלמיד יסביר מהו גרף דו צדדי ואת תכונותיו.
12. התלמיד יסביר מהי בעיית זיווג וזיווג מקסימלי.
13. התלמיד יסביר את האלגוריתם למציאת זיווג מקסימלי בגרף דו צדדי.

דגשים ודרכי הוראה

- המקרה הפשוט של צומת יעד (בור) יחיד.
- המקרה הלא פשוט של צומתי יעד (בורות) מרובים.
- המקרה הפשוט של צומת מקור יחיד.
- המקרה הלא פשוט של מקורות מרובים.
- המשמעות של אלגוריתם זרימה כאלגוריתם לסריקת גרף תוך קישור לבעיות סריקה שכבר הכירו בפרקים הקודמים בתכנית הלימודים – תחת אילוצי הקיבול. לדון במטרות של סריקה – לצורך חיפוש איבר, לצורך חישוב זרמים.
- וריאציה על הבעיה המקורית יכולה להוביל להגדרת זיווג מקסימלי לגרף דו צדדי ושימוש באלגוריתם פורד-פולקרסון למציאת פתרון עבור זיווג מקסימלי.

פרק 10: קידוד ודחיסת נתונים (8 שעות)

בעיה בסיסית: לצורך גיבוי של טקסט חשוב אנחנו מעוניינים לשמור אותו בצורה בינארית, ע"י כך שנקודת כל אות לרצף של ביטים. אבל על הקוד שניצור לקיים כמה תנאים:

1. יהיה אפשר לפענח את הקידוד באופן חד משמעי, כלומר, תמיד נדע לשחזר מתוך הקוד את הטקסט המקורי.
2. אורך הטקסט לאחר הקידוד יהיה קצר ככל שניתן.
3. ביצוע הקידוד והפענוח יהיו פשוטים ככל שניתן.

כדי שהקידודים שניצור יהיו קצרים ככל שניתן אנחנו נעזרים בידע שנחקר ונאסף על השפה בה הספר כתוב – נתונה לנו השכיחות של כל אות בשפה בתוך טקסטים הכתובים בשפה. כך למשל, בעברית לאותיות א', ה', י', ו' שכיחות גבוהה מאוד. לעומת זאת לאות צ' שכיחות נמוכה יחסית. הבעיה תיפתר ע"י האלגוריתם לבניית קוד הופמן.

מושגים

- קוד דחיסה
- קוד הופמן
- קוד תחיליות

מטרות ביצועיות

1. התלמיד יסביר מהו קידוד ומה זה קוד תחיליות.
2. התלמיד יסביר את האסטרטגיה מאחורי האלגוריתם של הופמן.
3. התלמיד יסביר את סיבוכיות זמן הריצה של האלגוריתם של הופמן.
4. התלמיד יסביר הקשר בין מספר העלים לבין מספר הצמתים שיש בעץ הופמן.
5. התלמיד יתכן אלגוריתם הפותר בעיה נתונה תוך שימוש באלגוריתם של הופמן כקופסה שחורה.
6. התלמיד יסביר איך גישה חמדנית באה לידי ביטוי באלגוריתם של הופמן ומדוע גישה זו מניבה פתרון אופטימלי למציאת קידוד שאורכו מינימלי.

דגשים ודרכי הוראה

- בניגוד לבעיות שנדונו בפרקים קודמים בעיית הקידוד שמוצגת כאן נראית רחוקה מגרפים ובכל זאת מסתבר שהשימוש בגרפים (בפרט, בעצים) מועיל לפתרונה.
- זו דוגמה לשימוש באסטרטגיה חמדנית.
- העץ הנוצר הוא מעניין משום שבניגוד למרבית העצים שהתלמידים נתקלו בהם בלימוד מדעי המחשב, כל המידע המשמעותי בעץ שמור אך ורק בעלים.

נושא רשות : מומלץ להרחיב את הנושא קידוד ודחיסת נתונים וללמד נוסף לאלגוריתם הופמן את האלגוריתם של למפל זיו.

