**ממן 12 – אלון גולדמן**

**שאלה 1**

1. נסמן בP את המסלול המזערי ובP' מסלול כלשהו אחר (שים לב שבשאלה הסימון הוא הפוך), לא הכי קצר. נניח בשלילה שעבור מסלול P' כל הצלעות שימושיות, אבל P=! P', ז"א הוא לא המסלול המזערי.

**טענת עזר**: יש בP' לפחות צלע אחת שלא קיימת בP.

הוכחה: נניח בשלילה שאין אף צלע בP' שלא קיימת בP. במצב כזה לא יתכן שיש צלעות שקיימות בP ולא קיימות בP', כי אחרת סכום המשקלים של P' היה קטן יותר מסכום המשקלים של P (כי לP' יש רק חלק מצלעות P בלי צלעות עודפות), וזה סותר את ההגדרה.

קיבלנו שלP' אין צלעות עודפות על P ולP אין צלעות עודפות על P', מכאן שP=P', וקיבלנו סתירה להנחה.

מכאן שהטענה נכונה – קיימת לפחות צלע אחת בP' שלא קיימת בP.

**טענת עזר שניה**: יש לפחות צלע אחת בP' שלא קיימת בP ומצביעה על צומת ששייכת למסלול P.

הוכחה בצורת אלגוריתם: נבחר צלע כלשהי X שלא קיימת בP (כמו שהוכחנו מקודם). אם היא מצביעה לצומת ששיכת לP – מצאנו. אם הצומת לא שייכת לP, אז נמשיך במסלול P' לצלע הבאה במסלול שאחרי X. הצלע הזאת גם לא שייכת לP (כי אחרת גם X היה שייך לP). נמשיך באיטרציה הזו עד שנמצא צומת ששיכת לP.

מכאן שהטענה נכונה – יש לפחות צלע אחת בP' שלא קיימת בP ומצביעה על צומת ששייכת למסלול P.

ניקח את הצלע הזו (נסמן בA)(ששייכת לP') ואת הצומת הזו (נסמן בX)(ששייכת גם לP' וגם לP). מכיוון שA לא שייכת לP, יש צלע כלשהי B ששייכת לP ומובילה גם היא לX.

**טענת עזר שלישית**: כל תת מסלול של מסלול מזערי, גם הוא מזערי

הוכחה: יהי P מסלול מינימלי כלשהו בין S לT (לא אותם S וT מהשאלה, אלא צמתים כלללים כלשהם). יהי P' תת מסלול מינימלי כלשהו של P, מX לY (X וY שייכים למסלול P והם הקצוות של P').

אם P' אינו מינימלי, אז יש מסלול קל יותר בין X לבין Y, וזה סתירה למינימליות של P.

הוכחנו ש כל תת מסלול של מסלול מזערי, גם הוא מזערי.

נבחן את המסלול מS לבין X. המסלול הקצר ביותר בין S לX הוא תת מסלול של P בהכרח, מכיוון שכל תת מסלול של מסלול מזערי הוא גם מזערי. מכיוון שזהו מסלול מזערי, אז צלע B היא שימושית. (מכיוון שהיא האחרונה במסלול המינימלי בין S לX).

תת המסלול של P' מS לX (דרך A) הוא לא מינימלי, ולכן A היא לא שימושית.

זה עומד בסתירה להנחה שלנו, שכל הצלעות בP' שימושיות, ולכן הנחת השלילה לא נכונה.

מכאן שאם במסלול כלשהו כל הצלעות שימושיות, המסלול הוא מזערי.

1. נניח שהצלע הלא שימושית (נסמן בA) מגיעה לצומת X. המסלול מS לX שעובר בA בהכרח לא מזערי, כי אחרת A היתה שימושית (לפי ההגדרה של צלע שימושית).

אם A לא שימושית אז יש בהכרח מסלול שמגיע לX דרך צלע שימושית אחרת, והוא הקצר ביותר מS לX.

ז"א שהמסלול המבוקש בשאלה מגיע לX דרך המסלול הלא מזערי, ואז ממשיך לV בדרך אחרת כלשהי. אם נחליף את המסלול מS לX במסלול המזערי, בהכרח נקבל מסלול יותר קצר. (כי הראינו שיש מסלול קצר יותר שעובר דרך צלע שימושית).

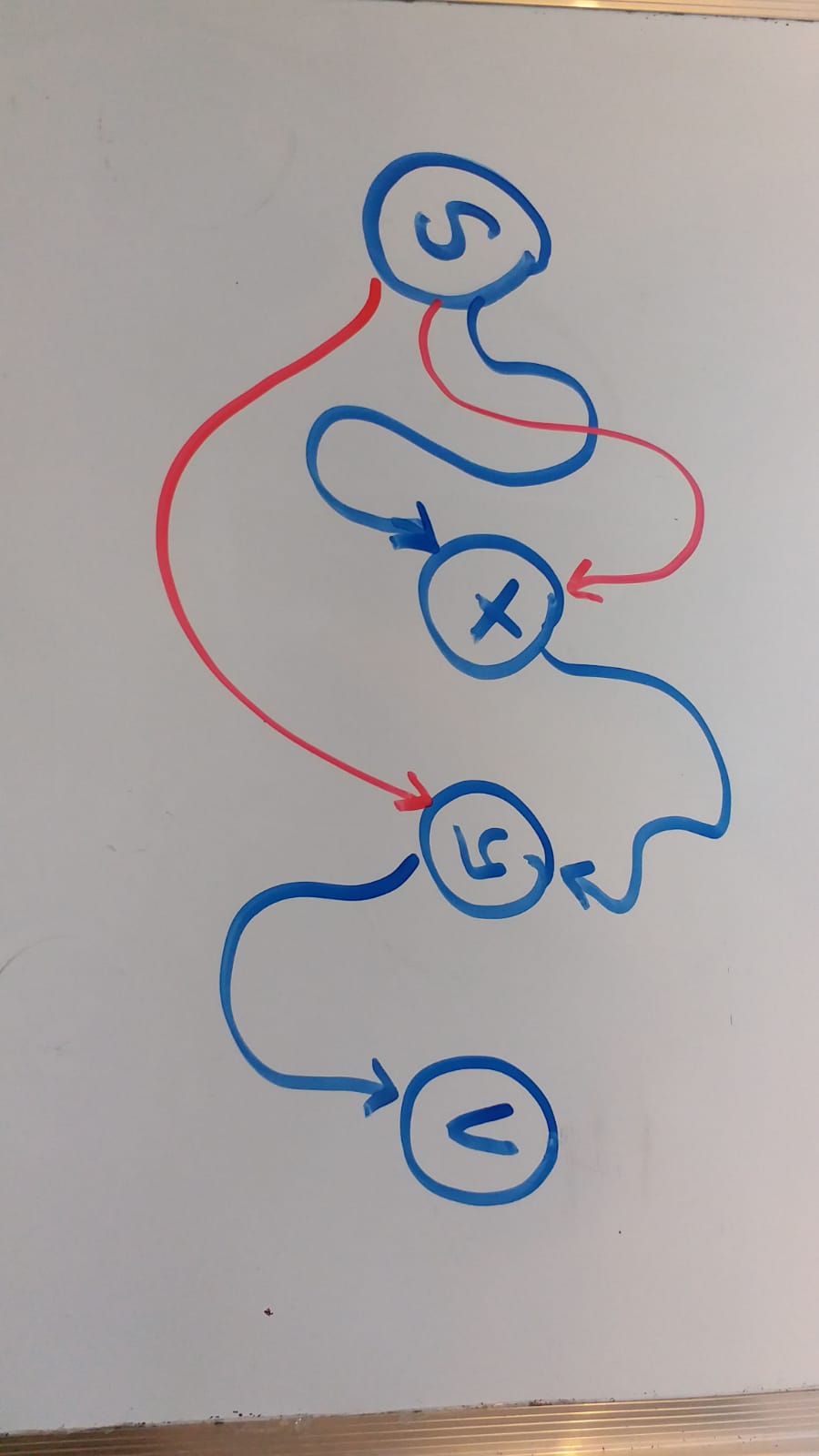
מכאן שהמסלול P שיש בו לפחות צלע אחת לא שימושית, אינו מזערי.

1. נוכיח שאם יש מסלול כמעט מזערי (נסמן בP), יש לו צלע לא שימושית אחת ויחידה. ז"א אין בו פחות מ1 צלעות שימושיות ואין בו יותר מ1.

אם היו במסלול פחות מצלע אחת שימושית, אז כל הצלעות במסלול היו שימושיות ואז לפי סעיף א' המסלול היה מזערי, וזה סותר את ההנחה. ולכן לא יכולות להיות במסלול כמעט מזערי פחות מצלע שימושית אחת.

נראה שיש רק אחת על דרך השלילה: נניח שיש מסלול כמעט מזערי ויש בו יותר מ1 צלעות שימושיות.

נראה דוגמה כללית של המסלול ונסביר:



הסבר על התרשים:

* המסלול הכחול מיצג את המסלול הכמעט מזערי.
* צמתים X וY מייצגים צמתים כלשהם במסלול, כאשר Y יכול להיות שווה לV, אבל X לא יכול להיות שווה לS.
* המסלולים מייצגים מסלול כלשהו (לאו דווקא ישיר)
* המסלולים האדומים יכולים להתמזג בכל שלב עם המסלול הכחול, מלבד העובדה שהם יכנסו לX ולY מצומת אחרת (ולא דרך המסלול הכחול). למשל, יתכן שהמסלול האדום לY מתמזג עם המסלול הכחול עד X ורק לאחר מכן מתפצל. זוהי רק דוגמה לשם המחשת ההוכחה.

נתון שיש מסלול מS לכל צומת בגרף. אם יש רק מסלול 1 מS לצומת כלשהי, אז המסלול הוא בהכרח הקצר ביותר ולכן הקשת הנכנסת לצומת בהכרח שימושית. מכך נובע שאם צלע היא לא שימושית, אז חייב להיות מסלול אחר, קצר יותר, מS לצומת זו.

הנחנו שיש לפחות 2 צלעות לא שימושיות – בתרשים זה מדובר על הצלעות הנכנסות לX ולY. נניח שהצלעות האדומות הם השימושיות.

נתון שהמסלול הכחול הוא כמעט מזערי. ז"א שיש רק מסלול אחד שיותר קצר ממנו. נבחן את המקרה שלפנינו:

1. המסלול האדום מS לX בהכרח קצר יותר מהמסלול הכחול לX (כי אחרת הוא לא היה שימושי). ניקח את המסלול האדום מS לX, ומשם נמשיך במסלול הכחול. קיבלנו מסלול יותר קצר מP.
2. המסלול האדום מS לY בהכרח קצר יותר מהמסלול הכחול לY. (כאמור, המסלול יכול להתלכד עם המסלול הכחול בהתחלה, אבל הוא מגיע לY מקשת אחרת אחת לפחות). ניקח את המסלול מS לY דרך האדום, ואז את המסלול הכחול לV. קיבלנו מסלול קצר יותר מP.
3. 2 המסלולים שמצאנו לא זהים, כי בראשון נכנסו לY דרך המסלול הכחול ובשני נכנסו דרך המסלול האדום.

קיבלנו 2 מסלולים קצרים יותר מP, וזה סותר את ההנחה – שהוא כמעט מזערי (ולכן יש רק מסלול אחד יותר קצר).

מכאן שההנחה לא נכונה – לא יכול להיות מסלול כמעט מזערי עם יותר מצלע לא שימושית אחת.

1. בכל P יש רק צלע לא שימושית אחת. נתון שהצלע הלא שימושית היא u1,u2, ומכאן נובע שכל שאר הצלעות הן שימושיות – ובפרט הצלעות שמS לu1 והצלעות שמu2 לV.

הצלעות שמS לU1 כולן שימושית לS, ולכן לפי סעיף א' המסלול הוא מזערי.

הוכחה שהמסלול מU2 לV חייב להיות מסלול מזערי:

נניח בשלילה שהמסלול הוא לא מזערי. ז"א שקיימת בו לפחות צלע לא שימושית אחת. אבל לפי סעיף ג' בכל P יש רק צלע לא שימושית אחת – ואנחנו קיבלנו 2 (הצלע הזאת, בנוסף לe). ולכן ההנחה לא נכונה, והמסלול מזערי.

1. **רעיון האלגוריתם**: נריץ את אלגוריתם דייקסטרה פעם אחת כדי ללמוד על סוגי הצלעות (שימושיות או לא), לאחר מכן נבנה גרף חדש ונריץ את אלגוריתם דייקסטרה שוב.

**הסבר**: במסלול כמעט מזערי יש צלע אחת שהיא לא שימושית, וכל שאר הצלעות שימושיות. לכן, נרצה "לכפות" על המסלול שלנו לכלול צלע לא שימושית, ובכך למצוא את המסלול כמעט הכי קצר.

נבצע שינוי קטן באלגוריתם של דייקסטרה – כאשר מוסיפים צומת מסויימת לרשימת הצמתים שנחקרו כבר, נסמן את הצלע המובילה אליה כשימושית.

לאחר מכן, נבנה גרף חדש, בצורה הבאה:

* כל צומת בגרף המקורי תופיע פעמיים בגרף החדש. לשם הנוחות, נקרא לקבוצה אחת של צמתים הקבוצה השמאלית, ולקבוצה השניה הקבוצה הימנית – כך שכל צומת מופיעה פעם אחת בקבוצה הימנית ופעם אחת בקבוצה השמאלית
* עבור כל הקשתות השימושיות בגרף המקורי, נבנה 2 קשתות בגרף החדש. קשת אחת שמחברת בין הצמתים בקבוצה השמאלית, וקשת אחת עבור הימנית.
* עבור כל קשת לא שימושית בגרף המקורי, נחבר בין צמתים משמאל לימין. למשל, אם בגרף המקורי היתה קיימת קשת כלשהי מצומת X לY, אז בגרף החדש תהיה קשת מצומת X השמאלית לצומת Y הימנית.
* אין שינוי למשקלי הקשתות

לאחר מכן נריץ את האלגוריתם של דייקסטרה שוב – הפעם בגרסה הרגילה – כאשר צומת המוצא היא הצומת S בגרף השמאלי, וצומת היעד היא הצומת t בגרף הימני. נקבל את המסלול כמעט מזערי.

**האלגוריתם**:

1. סמן את כל הקשתות כלא שימושיות
2. בצע את אלגוריתם דייקסטרה, עם השינוי המתואר.
3. בנה את הגרף החדש
4. הרץ שוב אל אלגוריתם דייקסטרה, הרגיל

**נכונות**:

כל הצמתים והקשתות בגרף החדש קיימות גם בגרף המקורי. לא הוספנו אף קשת שלא קיימת בגרף המקורי. ז"א שכל מסלול שנעשה בגרף החדש, אפשרי גם בגרף הישן.

נראה שהמסלול שקיבלנו הוא מסלול כמעט מזערי:

המסלול שנקבל יהיה בנוי בצורה הבאה:

1. מסלול כלשהו בתת הגרף השמאלי
2. קשת לא שימושית שמחברת מהגרף השמאלי לימני
3. מסלול כלשהו בתת הגרף הימני שמסתיים בT.

מסלולים 1+3 מורכבים אך ורק מקשתות שימושיות, מכיוון שאין קשתות לא שימושיות בתתי הגרפים (אלא רק ביניהם). קיבלנו מסלול שיש בו קשת לא שימושית אחת בדיוק.

בסעיף ג' הראנו שאם מסלול הוא כמעט מזערי, יש לו צלע לא שימושית אחת. זאת אומרת שהאפשרויות היחידות למסלולים כמעט מזעריים הם כל המסלולים האפשריים מS השמאלי לT הימני.

הגרף החדש ניתן לקבל רק מסלולים עם צלע אחת לא שימושית. מנכונות האלגוריתם של דייקסטרה נובע שהמסלול שמצאנו הוא הכי קצר מבין כל האפשרויות – ומכאן שהוא מסלול כמעט מזערי.

**סיבוכיות** (לפי המספור של שלבי האלגוריתם):

1. מעבר על כל הקשתות – O(m)
2. השינוי באלגוריתם דייקסטרה הוא שינוי בזמן קבוע (סימון קשתות), ולכן הסיבוכיות היא O(mlgn)
3. כל צומת מופיעה פעמיים, וכל קשת מופיעה לכל היותר פעמיים. לכן הבניה אורכת

1. האלגוריתם של דייקסטרה אורך אותו זמן כמו 2, רק עם שינוי בקבועים (פי 2 צמתים+קשתות), ולכן הסיבוכיות נשארת O(mlgn).

קיבלנו בסך הכל O(mlgn).

**שאלה 2**

**רעיון האלגוריתם**: נבצע סריקה של כל הקשתות שאלא נמצאות בT, אחת אחת, ונמצא מהן את הקשת המינימלית שתיצור את T'.

**הסבר**: T הוא עץ פורש מזערי של גרף G. השמטנו את אחת מצלעותיו, ונותרנו עם 2 חלקים של העץ. ראשית נמצא חלקים אלו ונפריד אותם.

נתון שהגרף הנותר G' קשיר, ז"א שיש צלעות אחרות המחברות בין 2 רכיבי הקשירות של הגרף. מבין צלעות אלו, נרצה להוסיף צלע אחת שתיתן לנו את המשקל הקל ביותר, ותחבר את 2 רכיבי הקשירות.

נעשה ז"א ע"י שנעבור על כל הקשתות שאינן חלק מT, ונבדוק עבור כל אחת אם היא מחברת את הרכיבים. מבין הקשתות שמחברות את הרכיבים, ניקח את המינימלית. בכל נקבל שוב עץ פורש, שיהווה את העץ הפורש המינימלי לG'

**האלגוריתם**:

נקודת המוצא של האלגוריתם היא שיש לנו 2 רכיבי קשירות עם קשתות ביניהם, כאשר בכל אחת מהם אנו יודעים מהם הקשתות השייכות לעץ הפורש המינימלי.

1. עבור 2 הצמתים שהיו בקצוות של הצלע e\*:
   1. הוסף את הצומת לרכיב הקשירות
   2. עבור לצמתים הבאים **על קשתות העץ**
2. עבור כל צלע שאינה בעץ:
   1. אם הצלע לא מחברת בין 2 רכיבי הקשירות השונים – עבור לצלע הבאה
   2. אם הצלע קלה יותר מהצלעות שנתקלנו בהן – שמור אותה
3. הוסף את הצלע המתקבלת לT'
4. החזר את T'

**נכונות**:

נוכיח את הטענות הבאות:

1. אחרי הסרה של צלע אחת e\*, נותרנו עם 2 רכיבי קשירות בדיוק.
2. רכיבי הקשירות המתקבלים הם עצים פורשים מינימליים כל אחד לרכיב הקשירות שלו.
3. הצלע שנוסיף תיצור עץ פורש מינימלי לG'.

הוכחה:

1. קשת מחברת בין 2 צמתים בלבד. לכן כל קשת יכולה לחבר בין 2 רכיבי קשירות ולא יותר. באופו האופן, לא יתכן שניתוק של קשת אחת יצור 3 רכיבי קשירות (כי אז חיבור שלה אמור לחבר 3 רכיבי קשירות, וזה לא אפשרי מעצם ההגדרה).

נניח שניתוק הקשת e\* יותיר אותנו עם רכיב קשירות אחד. זה אומר שיש בו לפחות n-1 קשתות. מכאן שאם נוסיף את e\* נקבל גרף עם n קשתות – אבל קיבלנו בחזרה את T ונתון שהוא עץ, ולעצים יש בדיוק n-1 קשתות. התקבלה סתירה, ומכאן שחייבים להיות 2 רכיבי קשירות כאשר נסיר את e\*.

1. נבחן את אחד מרכיבי הקישורת (ההוכחה תקפה גם לשני באותה מידה):

נניח בשלילה שהוא לא עץ פורש מינימלי. ז"א שקיים עץ פורש אחר שהוא המינימלי – נסמנו בT''.

ז"א ש:

נתון שT עף פורש מינימלי, ז"א שסכום המשקלים:

אם נחבר את העץ המינימלי שקיבלנו, עם הקשת e\*, ועם הרכיב השני, נקבל עץ פורש אחר:

ז"א שקיבלנו עץ יותר קל מT – וזאת סתירה לכך שT הוא עץ פורש מינימלי.

לכן הוכחנו שרכיבי הקשירות אחרי הסרת e\* הם עצים פורשים מינימלים.

1. נבחן את כל הקשתות שנותרו ששייכות לG' ולא לT. חלק מהקשתות מחברות בין צמתים באותו רכיב קשירות – נוותר עליהם. נשארו עם קשתות שמחברות בין 2 רכיבי הקשירות. מתוכן, האלגוריתם בוחר אחת באקראי, ומחליף אותה רק כאשר הוא מוצא קשת כזו, עם משקל קל יותר. ז"א שמשקל הקשת הולך ויורד. בסוף האלגוריתם, נמצא את הקשת שמחברת בין רכיבי הקשירות ומשקלה הנמוך ביותר.

נבחן את מה שקיבלנו:

* 2 רכיבי קשירות שהם עצים פורשים מינימלים
* קשת שמחברת ביניהם, הקלה ביותר האפשרית

אם נחבר את 2 רכיבי הקשירות בעזרת הקשת נקבל עץ פורש של G'. אם נחבר את כל המשקלים ביחד, נקבל את המשקל הנמוך ביותר האפשרי – ולכן קיבלנו את T', עף פורש מינימלי של G'.

**סיבוכיות** (לפי שלבי האלגוריתם):

1. סורקים את כל העץ ומחלקים לרכיבי קשירות. יש m קשתות וm+1 צמתים, ולכן הסיבוכיות היא O(2m+1) = O(m)
2. עבור לכל היותר m קשתות, מבצעים:
   1. בדיקת רכיבי הקשירות יכולה להתבצע בזמן קבוע ע"פ המימוש המתואר בספר לפעולת find בעמוד 165 (אין צורך בפעולת union ולכן מדובר בזמן קבוע ולא לוגריתמי)
   2. זמן קבוע

ז"א בסך הכל O(m)

3+4: זמן קבוע

קיבלנו בסך הכל O(m), כנדרש.

**שאלה 3**

מספיק שעבור פסוקית אחת, יהיו עבור כל 3 המשתנים שבה יותר פסוקיות אחרות (שטרם סופקו) עם המשנה ההפוך, וכך אחרי 3 איטרציות של האלגוריתם החמדן נקבל פסוק שכולו שקר.

נבנה פסוק כזה לדוגמה:

האלגוריתם ירוץ תחילת על x1. מכיוון שפסוקיות 2+3 יסופקו ע"י x1=F, (לעומת פסוקית 1 שתסופק ע"י ההפך), נשים x1-F.

עבור x2 נקבל את פסוקיות 4+5, לעומת פסוקית 1 – ולכן נציב גם x2=F.

עבור x3 נקבל את פסוקיות 6+7 לעומת פסוקית מס' 1, ולכן נציב גם 7 לעומת פסוקית מס' 1, ולכן נציב גם x3=F.

קיבלנו שפסוקית מס' 1 לא תסופק, והאלגוריתם יטעה ויחזית תשובה שלילית.

ניתן למצוא הצבה שכן תספק את הפסוקית, למשל עבור x1=T, x4=T.

**שאלה 4**

נוכיח בצורה קונסטרוקטיבית:

קלט: עף מושרש T בינארי לחלוטין

פלט: סדרת שכיחויות שאפשר לבנות ממנה את T

**הסבר**:

האלגוריתם אנו מנצלים את התכונות הבאות של עץ הופמן:

1. כל צומת בעץ מחזיק את הסכום של השכיחויות של 2 הבנים שלו
2. סכום כל השכיחויות הוא 1
3. כל העלים באותו עומק בעץ יכולים להיות בעלי אותה השכיחות (לפי עמוד 185 בספר)

בעצם, נבצע אלגוריתם הפוך לאלגוריתם של הופמן: האלגוריתם של הופמן מוצא בכל פעם את 2 השכיחויות הנמוכות ביותר, מחבר אותן, ומציב את התוצאה בצומת האב של 2 האחים עם השכיחויות הנמוכות.

האלגוריתם שלי יעבוד הפוך – השכיחות של השורש תהיה 1, ועבור כל הילדים שלו נחלק אותו ל2, באופן רקורסיבי. השכיחות של האב תהיה 1, של 2 בניו 0.5, של נכדיו 0.25 וכד'.

בצורה הזאת, ככל שמתרחקים מהשורש השכיחות נמוכה, כך שבעלים העמוקים ביותר נקבל את השכיחות הנמוכה ביותר.

**האלגוריתם**:

1. הצב בשורש שכיחות של 1
2. כל עוד לא הגעת לעלה:
   1. הצב ב2 הבנים את השכיחות של האב חלקי 2

האלגוריתם נכון מכיוון שהוא מקיים את התנאי לעץ הופמן במשפט 4.29 – אם יש עלה בעומק עמוק יותר מעלה אחר, אז גם השכיחות שלו נמוכה יותר. זה בוודאי נכון מהאלגוריתם, מכיוון שבכל רמת עומק השכיחות מתחלקת ב2.