גנומיקה חישובית עבודה 2

מתן בזן - 201085016

Q1-

אבחנה 1- מספר הצעדים של כל מסלול הוא n-m-i צעדים כאשר i הינו מספר האלכסונים (נובע מהעובדה שאורך מסלול מקסימאלי הינו באורך n+m צעדים וביצוע צעד אלכסוני חוסך צעד אחד).

אבחנה 2- מספר האלכסונים חסום על ידי אורך המילה המינימאלית כלומר המינימום בין m לn (כמובן גדול שווה מאפס).

$$\sum_{i=0}^{\min(n,m)} \binom{m+n-i}{i}\binom{m+n-2i}{n-i}$$

הסבר הנוסחה:

לכל i נבחר i אלכסונים מתוך n+m-i צעדים (אבחנה 1), כעת נשארו $(m+n-i)-i=(m+n-2i)$ צעדים לבצע בפרט n-i צעדים ימינה ו m-i צעדים למטה. נבחר בה"כ n-i צעדים ימינה מתוך הצעדים הנותרים (כאשר הצעדים שמאלה נקבעים מעצמם).
הסכימה של כל האפשרויות לבחור את i תיתן את מספר המסלולים כנדרש.

Q2-

1. על פי האלגוריתם ל splacing נוסחת DP לשורה הראשונה מוגדרת על ידי:

DP(i, j, B) =
max { *DP(i, j – 1, B) – indel penalty*
$\max_{\text{all blocks } B' \text{ preceding block } B}$ *DP(O(B'), j, B') – indel penalty*
$\max_{\text{all blocks } B' \text{ preceding block } B}$ *DP(O(B'), j – 1, B') + δ(g_i, t_j)*
}

נסתכל על כל בלוקים הקודמים האפשריים אלה הם הבלוקים העליון השמאלי והעליון הימיני (זאת משום שעל פי המיון אינם חופפים לאותו האקסון שאנו מחפשים).
כעת בהינתן שהאות הראשונה של השורה היא A נחשב את הערך עבור כל תא בעזרת נוסחאות לעיל, נשים לב כי עבור התא הראשון ניתן לבחור את הערך המקסימאלי מבין match עם כל אחד מן המקומות המתאמים של הבלוקים או אפס, ונקבל –

0	-2	-4	-1	0	1	-1

2. ננתח את זמן ריצת האלגוריתם ל spliced alignment-
כאמור בהינתן רצף T באורך n, b המסמן את סכום אורכי הבלוקים בקבוצת האקסונים הפוטנציאלים B, וk המסמן את מספר הבלוקים בקבוצת האקסונים הפונטנציאלים.

I. מיון האקסונים על פי אינדקס סיום של האינטרוולים $O(K^2)$
II. נחשב העמדה בהתחשב בסדר המיון בהתאם לנוסחאות הDP המתאימות עבור כל בלוק סה"כ $O(bn)$ על פי האלגוריתם של N&W אקסונים באורך b וT באורך n.
III. כמו כן בשורה הראשונה של כל בלוק סה"כ K בלוקים מבצעים בדיקה על מנת לחשב את הערכים ההתחלתיים לכל השורה ולכל תא אל מול כל שאר התאים המתאמים כאמור ישנם k כאילו וכן עבור כל אות במחרוזת T בארוך n.
כלומר אורך n עם k בלוקים k פעמים, סה"כ $O(nk^2)$.

בסה"כ קיבלנו $O(bn+nk^2)$ כנדרש.

גנומיקה חישובית עבודה 2

מתן בזן - 201085016

Q3-

0.

	A	B	C	D	E
A		24	28	32	36
B			16	20	24
C				8	12
D					16
E					

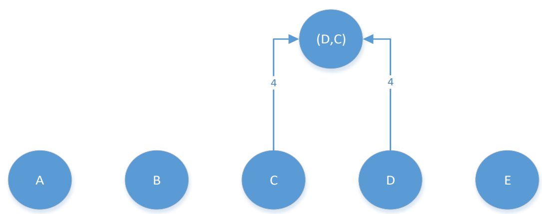
1.

$$D_{(d,c),a} = \frac{1}{2}(1 \cdot 28 + 1 \cdot 32) = 30$$

$$D_{(d,c),b} = \frac{1}{2}(1 \cdot 16 + 1 \cdot 20) = 18$$

$$D_{(d,c),e} = \frac{1}{2}(1 \cdot 12 + 1 \cdot 16) = 14$$

	A	B	(D,C)	E
A		24	30	36
B			18	24
(D,C)				14
E				

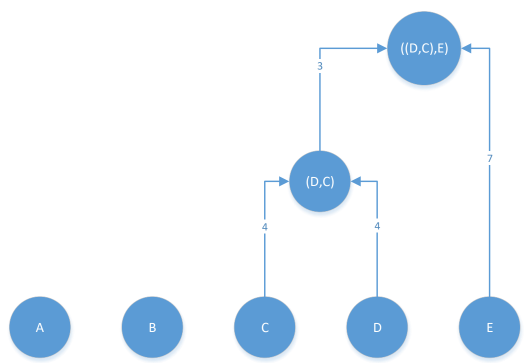


2.

$$D_{((d,c),e),a} = \frac{1}{3}(2 \cdot 30 + 1 \cdot 36) = 32$$

$$D_{((d,c),e),b} = \frac{1}{3}(2 \cdot 18 + 1 \cdot 24) = 20$$

	A	B	((D,C),E)
A		24	32
B			20
((D,C),E)			

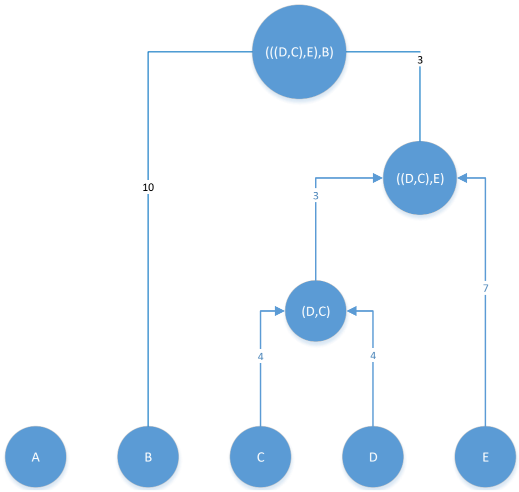


גנומיקה חישובית עבודה 2
מתן בזן - 201085016

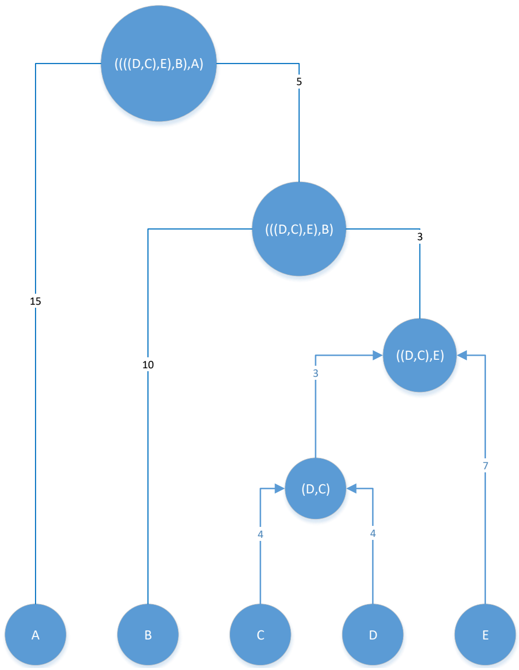
.3

	A	(((D,C),E),B)
A		30
(((D,C),E),B)		

$$D_{(((d,c),e),b),a} = \frac{1}{4}(3 \cdot 32 + 1 \cdot 24) = 30$$



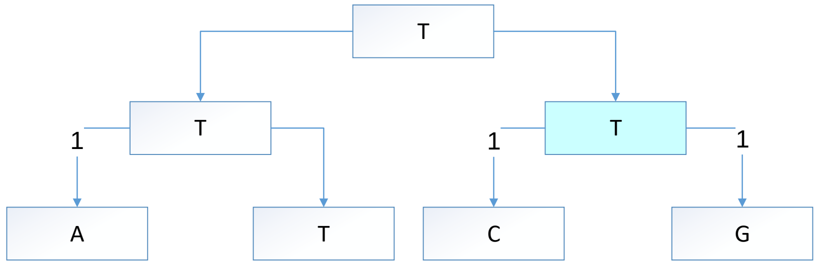
.4



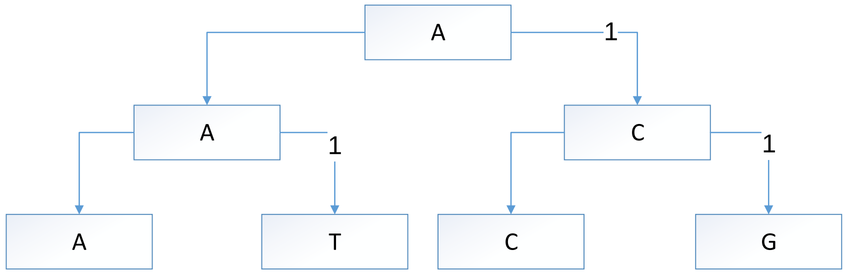
גנומיקה חישובית עבודה 2
מתן בזן - 201085016

-Q4

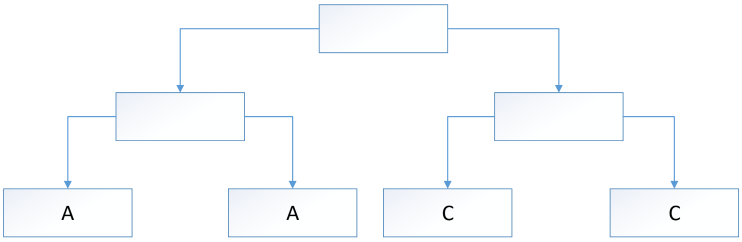
.1 ראשית נשים לב כי העץ הנ"ל אינו שייך לN שכן הצומת המסומן לא יכול להיבחר כחלק מהאלגוריתם של fich שכן החיתוך או האיחוד של שני הבנים שלו אינם מכילים אותו.

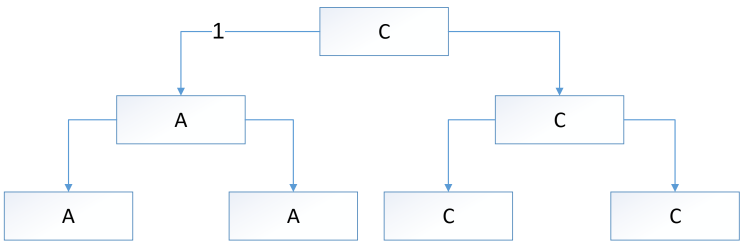


כעת נראה כי זהו עץ בעל parsimony score אופטימאלי, על ידי הפעלת האלגוריתם של fich שכן אם אכן התוצאה זה, זהו עץ אופטימאלי, עובדה הנובעת מכך שהאלגוריתם של fich מחזירה עץ בעל ניקוד אופטימאלי.
כפי שניתן ראות מהעץ שלמטה אכן parsimony score בשני העצים, כלומר העץ לעיל שייך לM ואינו שייך לN.

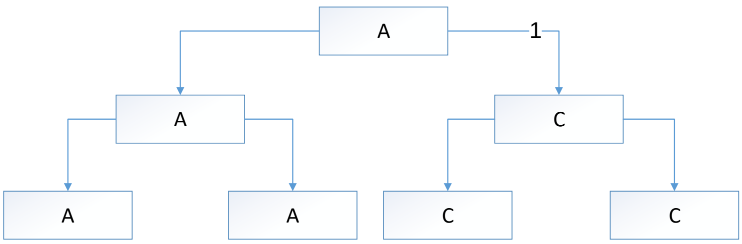


.2 נראה השמה לעליי העץ כך שאלגוריתם fich יוכל להחזיר יותר מעץ אופטימאלי יחיד.





עץ א

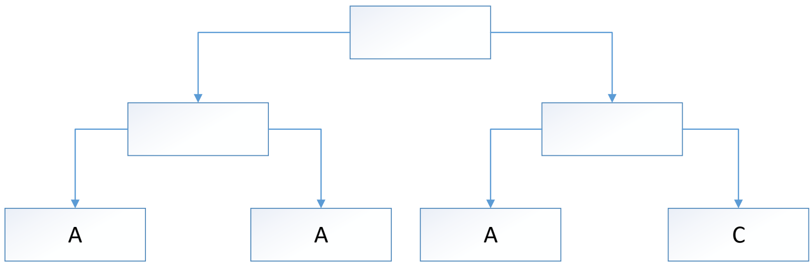


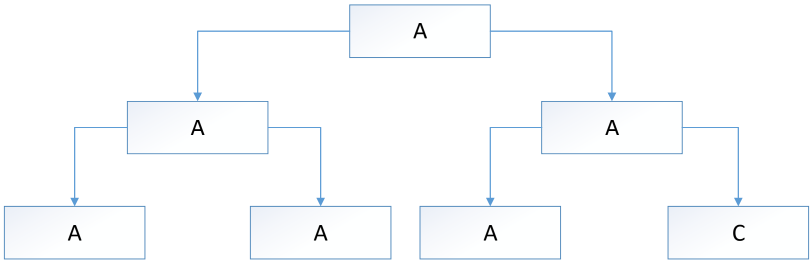
עץ ב

גנומיקה חישובית עבודה 2

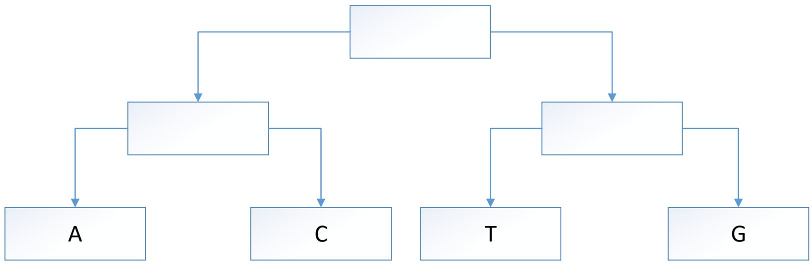
מתן בזן - 201085016

3. עבור ההשמה הנ"ל הקבוצה N בגודל אחד בלבד-





4. עבור ההשמה הבאה הציון המתקבל הינו הציון האופטימאלי הגרוע ביותר שאפשר לקבל בהינתן עץ בינארי מלא מעל ארבע עלים.



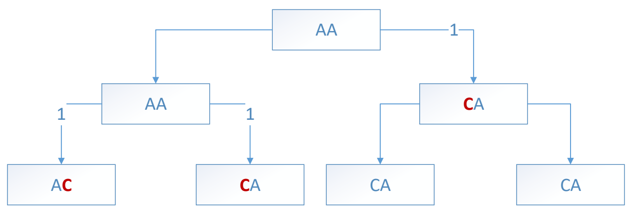
אנו יודעים כי התוצאה של האלגוריתם של FICH יפיק היא הסכום של 1 עבור כל מוטציה של אות. כאשר מסתכלים על עץ בינארי מלא עם 4 עלים של פתרון אופטימלי, השורש חייב להיות אחת האותיות מן העלים, והמספר המינימלי של מוטציות בעץ הנ"ל הוא 3 לכל היותר, שכל צומת אב הוא בדיוק אחת מהאותיות של הבנים שלו כך שנקבל כי בשכבה התחתונה ישנם לכל היותר שני מוטציות ובשכבה השנייה לכל היותר אחת, ובסה"כ לכל היותר שלוש מוטציות.
מכאן שזהו העץ האופטימאלי עם הציון הגרוע ביותר שניתן לקבל.

גנומיקה חישובית עבודה 2

מתן בזן - 201085016

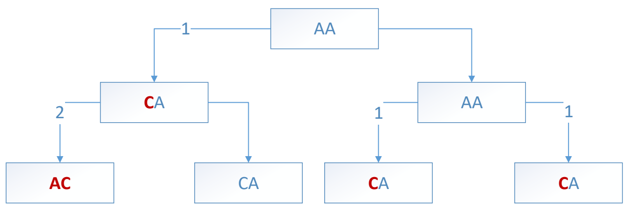
Q5-

1. על פי עיקרון parsimony לעץ א' ניקוד נמוך יותר של 3 מאשר עץ ב' 5 , ולכן יועדף על פני עץ ב'.



עץ א

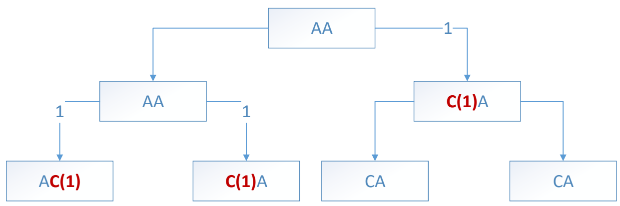
Scoring parsimony 3



עץ ב

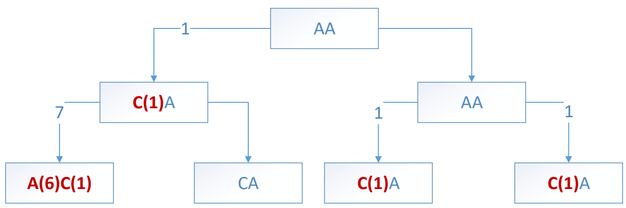
Scoring parsimony 5

2. על פי עיקרון parsimony לעץ א' ניקוד נמוך יותר של 3 מאשר עץ ב' 10, ולכן יועדף על פני עץ ב'.



עץ א

Scoring parsimony 3

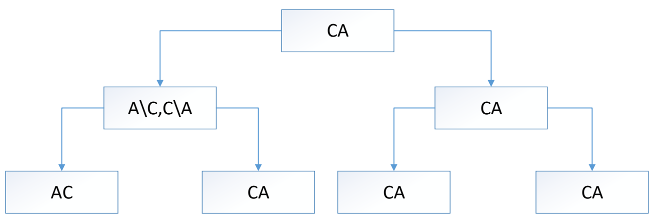


עץ ב

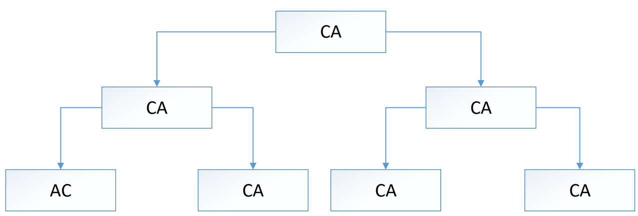
Scoring parsimony 10

3. נציג חישוב העץ המינימאלי תוך שימוש באלגוריתם של fitch.

שלב א



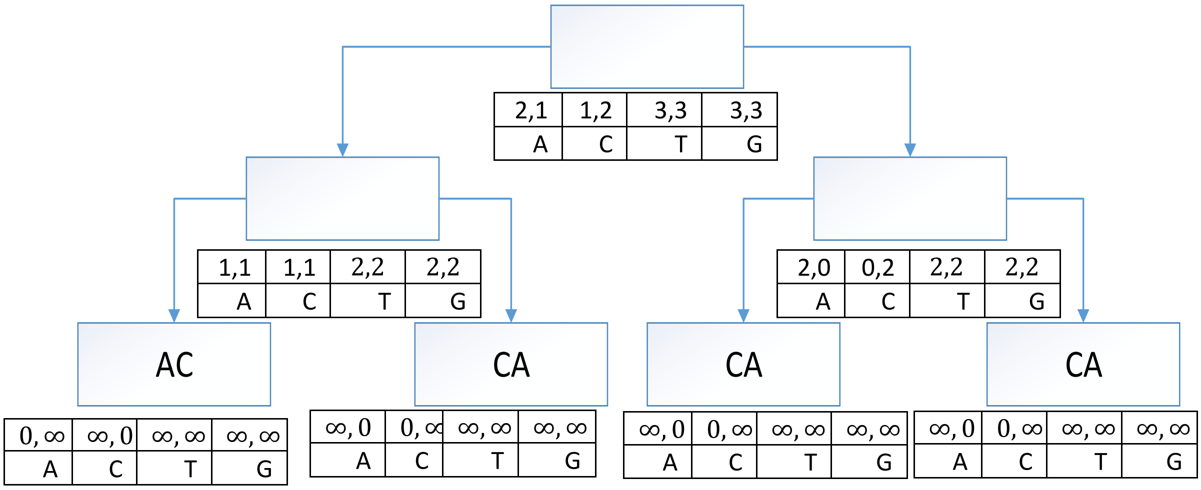
שלב ב



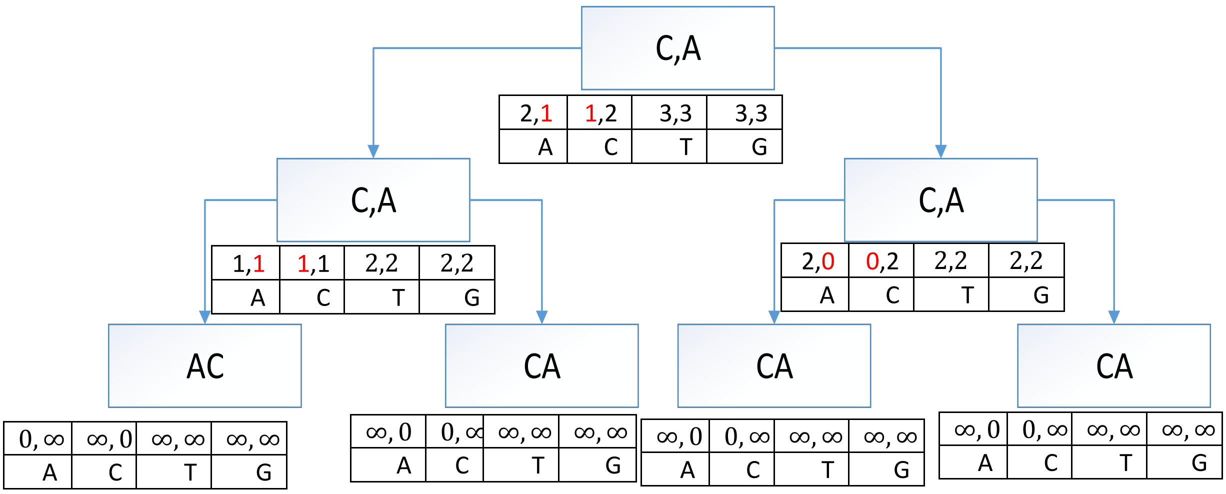
גנומיקה חישובית עבודה 2

מתן בזן - 201085016

4. Sankoff’s buttom up-



Sankoff’s top down-



גנומיקה חישובית עבודה 2

מתן בזן - 201085016

5. כפי שניתן לצפות parsimony score בהרצת שני האלגוריתם זהה ושווה ל2,בהינתן מטריצת הניקוד הנתונה.
כעת נערוך השוואה בסיסית בין האלגוריתם של Fitch לבן זה של Sankoff-
   I. ראשית כמובן שני האלגוריתם נותנים פתרון לבעיה זהה של חישוב עץ האבולוציה שנותן מינימום parsimony score.
   II. זמני הריצה של האלגוריתם זהים O(NK) כאשר N הוא מספר העלים וK הינו מספר האותיות.
   III. בהינתן מטריצת הניקוד לעיל, שבו התאמה מקבלת 0 ו חוסר התאמה מקבלת 1, האלגוריתם של Sankoff והאלגוריתם של Fitch מתנהגים אותו דבר, במובן ששניהם מייצרים את אותו סט של פתרונות אופטימליים.
   עובדה זאת נובעת מכך שעבור האלגוריתם של Sankoff, עבור צומת מסוים מהנוסחה האות שתמשיך למעלה במהלך ריצת הbuttom up הינה האות עם הערך המנימאלי בתוך אותה הקבוצה שבצומת. מכאן בהינתן צומת N עם ילדים X וY,אם קיים איבר משותף בקבוצה X ובקבוצהY, ולכן גם שייך לקבוצת הצומת N.
   אחרת כלומר אם אין איברים משותפים הקבוצה של צומת N צריכה להכיל את כל האותיות משתי הקבוצות הבנות X וY, וקיבלנו בדיוק את הנוסחה של האלגוריתם של Fitch.

$$S_N = \begin{cases} S_y \cap S_x & , S_x \cap S_y \neq \phi \\ S_y \cup S_x & else \end{cases}$$

   מכאן נסיק כי האלגוריתם Fich הינו מקרה פרטי של האלגוריתם של Sankoff.

6. קיים דרך להרחיב את האלגוריתם של sankorff כך שיאפשר החזרה של כל הפתרונות האופטימליים.
נתאר את הבנייה אשר תעזור לנו למצוא את הפתרונות בהתבסס על האלגוריתם של sankorrf-

בדרך דומה של האלגוריתם נבצע שני שלבים buttom up וtop down, על מנת למלא את מבנה הנתונים שיאפשר לנו לשחזר את הפתרונות.

Bottom up-

נבצע בדיוק כמו האלגוריתם הרגיל של sankorff, נתבסס בשלב הבא על המידע שנאסף בצמתים על מנת למצוא את התיוג האופטימאלי האפשרי של הצמתים.

Top down-

כעת שסיימנו את חישוב כל הצמתים, נבצע שחזור של כל העצים האופטימליים.
נתחיל משורש העץ ,נשים לב כי כל אחד מהאותיות בעל ניקוד מינמאלי הינו שורש של עץ אופטימאלי חוקי, ועל כן ניצור קבוצה S כך שאות שייכת לעץ אם היא מינימאלית בשורש.

כעת עבור כל אות בקבוצה זו, ניצור עץ אשר קוד' השורש שלו מתויג באות הנ"ל, נמשיך בצורה רקורסיבית עד העלים בצורה הברה:

- עבור כל אות C נבחן את כל האפשרויות של צימוד של הבנים שלו כך שציון הזוג ייתן את הציון המיניאלי של צומת C , ולכן יכול להיות חלק מן הפתרון האופטימאלי.
- כעת שמצאנו זוג שכזה נשכפל את העץ עם, עד אותו הקוד' מהשורש ונוסיף את הבנים שמצאנו, נשיך בתהליך זה ובדרך זו נקבל לבסוף את קבוצת העצים האופטימאליים.

כלומר בסופו של דבר אנו בודקים האם יכלנו לייצר את הציון האופטימאלי מכל זוג בנים ובכך לבנות את הפתרון.

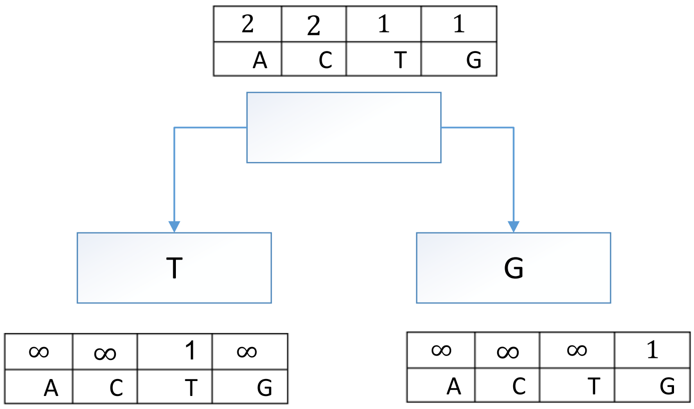
זמן הריצה של השלב הראשון bottom up זהה לזה של sankorff, השלב השני בהנחה שבכל צומת מס' האותיות חסום, אזי מציאת זוג הינו ליניארי בגודל זה כמו כן נחזר על

גנומיקה חישובית עבודה 2

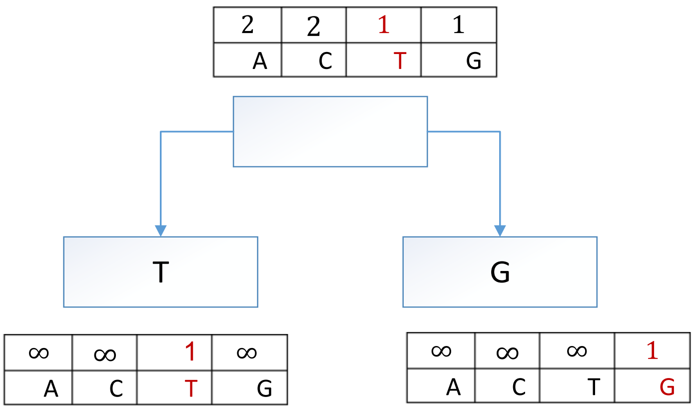
מתן בזן - 201085016

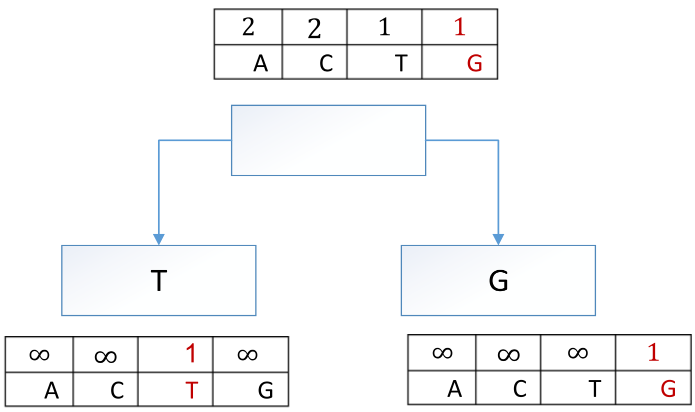
התהליך כמספר הצמתים ולבסוף בכל שלב נבצע העתקה של לכל היותר מס' הצמתים כך שסה"כ זמן השחזור והריצה ליניארי כגדול הקלט.

דוגמאת ריצה עם טבלת המשקולות מסעיף 4,שלב ה bottom up-



שלב הtop down-





גנומיקה חישובית עבודה 2

מתן בזן - 201085016

7. קיים דרך להרחיב את האלגוריתם של Fitch כך שיאפשר החזרה של כל הפתרונות האופטימליים.
נתאר את הבנייה אשר תעזור לנו למצוא את הפתרונות, האלגוריתם של Hartigam-

Buttom up-

לכל צומת, נאסוף את המועמדים להיות חלק מן הפתרון האופטימאלי, המועמדים יישמרו בשתי קבוצות קבוצה S_1 , S_2 כאשר קבוצה 1 הינה הקבוצה שהאלגוריתם של fitch מחשב (כלומר הצעד הרגיל) והקבוצה 2 הינם קבוצת מועמדים נוספת (להיות חלק מהפתרון האופטימאלי) כאלה שהאלגוריתם לא בחר.

נגדיר כללים עבור חישוב קבוצה 2-

1. עבור עלים- הקבוצה הינה הקבוצה הריקה.
2. עבור צמתים- נסמן בN את המספר המרבי של הופעת האות בין הצמתים בעץ. כעת לכל צומת פנימי V (שאינו עלה), אם ולכל אות אם מספר המופעים שלה בתת העץ של v, הוא שווה לN-1 נכניסו לקבוצה 2.

For each state b, define $k(b) := \left|\{v_i \mid b \in \mathrm{S}_1(v_i)\}\right|$
$K := \max_b\{k(b)\}$
$\mathrm{S}_1(u) := \{b \mid k(b) = K\}$
$\mathrm{S}_2(u) := \{b \mid k(b) = K-1\}$

Top down שחזור הפתרונות-

כעת נצטרך להחליף בכל מצב (צומת) איזה מבין האותיות יהיה חלק מהפתרון האופטימאלי.

נחלק למקרים-
עבור השורש- בחירה בכל אחד מהאותיות בקבוצה 1 יהיה חלק מפתרון אופטימאלי.

עבור שאר הצמתים מלמעלה למטה, נעבור על הצמתים ונקצה תיוגים בהתאם לתיוג האב V בצורה הבאה-

- אם אות שייכת לקבוצת האב מס' 1 וגם לקבוצה הבן מס' 1 אזי ניתן לבחור אות זו שכן אינה מוסיפה ניקוד וכן חלק מפתרון אופטימאלי.
- אם אות שייכת לקבוצת האב מס' 1 וגם לקבוצת הבן מס' 2 אזי בכל מקרה נצטרך לבזבז 1 (ניקוד) עד הפתרון האופטימאלי ולכן נוכל לבחור באות זו כחלק מהפתרון האופטימאלי.
- אם האות שייכת לקבוצת האב 1 ואינה שייכת לאף אחת מקבוצת הבן, ניבחר באות אשר תיתן לנו את מספר המוטציות (ציון) הנמוך ביותר, במילים אחרות אנו משלמים 1 לכל מוטציה מהן לאב, נרצה להביא למינימום את העלות ולכן נבחר מקבוצה 1 של הבן אשר מופיע הכי הרבה פעמיים בתתי העץ שלו.

$$\text{for a given parent node } p \text{ and a child node } c, \text{for each } a \in S_1(p),$$
$$Sol(c,a) = \begin{cases} \{a\}, & if\ a \in S_1(c) \\ \{a\} \cup S_1(c), & if\ a \in S_2(c) \\ S_1(c) & otherwise \end{cases}$$

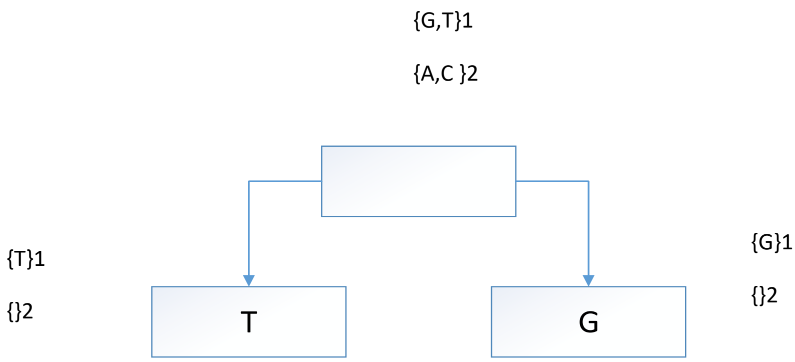
נשים לב כי הבנייה מתבססת על הרצת האלגוריתם של Fitch ואינה מוסיפה לזמן ריצתו.

על מנת לשחזר את הפתרונות האופטימליים לאחר הבנייה כל שנעשה הוא להתחיל מהשורש, ולחבר את כל הפרמוטציות על פי השייכות לקבוצת הפתרונות כפי שהגדרנו, בבניה לעיל, כל זו בזמן ריצה לינארי לאורך הפתרון כלומר נעבור בכל צומת במסלול פעם אחת.

גנומיקה חישובית עבודה 2

מתן בזן - 201085016

דוגמא עבור העץ הבא שלב ראשון Buttom up-



כעת נדגים את שלב בחירת העצים מלמעלה למטה עבור כל בחירה לפי האלגוריתם והבחירה נשים לב שהעץ האופטימאלי הוא עם ניקוד 1.