**רשתות תקשורת מחשבים**

**תרגיל תיאורטי 2#**

## **מגישים:**

## שם: אבי קצ'ולרו

## ת.ז. 203056585

## מייל: [avi.c33@gmail.com](mailto:avi.c33@gmail.com)

## שם: נעם גוטליב

## ת.ז. 201606951

## מייל: [noam.got@gmail.com](mailto:noam.got@gmail.com)

## שם: מתן סירי

## ת.ז. 304957673

## מייל: [matanse@gmail.com](mailto:matanse@gmail.com)

# שאלה 1

1. The root of the network is bridge 6 (because it has the smallest ID).

|  |  |  |
| --- | --- | --- |
| Bridge | Root port | Designated ports |
| 6 | - | 1 |
| 11 | 1\* | - |
| 19 | 1 | 2, 3 |
| 23 | 2 | 1 |
| 35 | 2 | 1 |
| 42 | 2 | 1 |

\*note that since bridge 11 has no designated ports, its root port is blocked and therefore this bridge should be removed from the network.

1. Yes we can!

In the current configuration, the path of the message is:

C 🡪 19 🡪 A 🡪 23 🡪 F 🡪 42 🡪 E

By changing bridge 11 ID to 4 (for example; any ID smaller than 6 would work), the new ST root would be 4. In that case, when sending a message from C to E we can get a shorter message path:

C 🡪 35 🡪 D 🡪 4 (originally 11) 🡪 E

Namely, the message goes through 2 bridges instead of 3 (in the original configuration), so indeed we get an improvement. Great success!

# שאלה 2

1. **Claim**: after the STP protocol converges a bridge does not have a root port iff this bridge is the root.

**Proof:**

* Let B be the ID of the root bridge in the network. Let us assume that B is connected to N neighbors (bridges) with IDs B1, … , BN. According to the protocol, each bridge of B1, … , BN will get the following message from B: **B.0.B**. This message will "beat" any of the other messages gotten from its neighbors (according to the lexicographic order we defined in class). For example: if some (neighbor) bridge Bi sends the following message: **root\_id.len.Bi** then B ≤ *root\_id* because B is the root, and therefore it holds the smallest ID in the network (this is why it was chosen to be the root in the first place…). If B = root\_id then it must hold that 0 < *len*, because in that case Biconsiders B as the root port, and therefore it knows that the length from it is at least 1 (len = 0 only if Bi thinks it is the root, and that is not the case anymore). All in all we get that B sent a better message than its neighbors, and therefore (as we saw in class) it has no root port.
* Let B be the ID of a bridge in the network which is not the root bridge. Let B\* be the ID of a bridge that is a neighbor of B and has the shortest distance to the root bride among B's neighbors. W.L.G we will also assume that if there are more neighbors of B with the same distance from the root, their IDs are larger than B\*. When the algorithm converges, the last message sent from B\* would be **root\_id.len.B\*** where *root\_id* and *len* are the **real** root ID and the optimal distance from B\* to it, respectively. Since B\* has the lowest ID among B's neighbors with the minimal distance from the root, then B will choose to send its data through B\* (namely, the message B gets from B\* beats the other messages known to B, including its own message), which will make B\* its root port – so indeed it has one. Notice that this proof still holds in the case where B\* is the root.

1. נניח בשלילה כי קיים port שהbridge מגדיר אותו להיות גם designated וגם root:

נסמן את ההודעה הטובה ביותר שקיבל הbridge ב. הודעה זו מקיימת שלכל הודעה אחרת, מתקיים . עבור כזה, תהא ההודעה: הודעה פחות טובה מ-. שהתקבלה על הפורט הנתון, לאור הנתון שהפורט הינו designated. מפני שהport הוא גם root לפי ההנחה, נקבל כי הודעה זה מקיימת שלכל הודעה אחרת, , מתקיים ש-, בפרט עבור , בסתירה למינימליות של .

# שאלה 3

נוסיף עוד שדה להודעה שיהיה השדה הרביעי שלה, שיציין את מס' הport ממנו הגיעה ההודעה. ההשוואה תתבצע כמו בSTP המקורי, כלומר שדה-שדה באופן לקסיקוגרפי, וכך בהינתן כמה הודעות, נבחר את השדה הרביעי בהודעה הטובה ביותר להיות הdesignated port (בהינתן כמובן שהbridge איננו root), ואת שאר הportים ניתן להגדיר כblocking.

# שאלה 4

מצב התחלתי:

10

100

100

12

12

15

100

100

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| חיבור | תחנת מקור | תחנת יעד | מסלול | ר"פ מבוקש | ר"פ זמני |
| 1 | A | C | wy | 8 | 0 |
| 2 | B | C | xzy | 19 | 0 |
| 3 | A | D | wxz | 15 | 0 |
| 4 | A | D | wxz | 22 | 0 |

נוכל להוסיף רוחב פס 3.33 לכל החיבורים במידה שווה:

**10/10**

**3.33/100**

**10/100**

**3.33/12**

**6.67/12**

**3.33/15**

**6.67/100**

**6.67/100**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| חיבור | תחנת מקור | תחנת יעד | מסלול | ר"פ מבוקש | ר"פ זמני |
| 1 | A | C | wy | 8 | **3.33** |
| 2 | B | C | xzy | 19 | **3.33** |
| 3 | A | D | wxz | 15 | **3.33** |
| 4 | A | D | wxz | 22 | **3.33** |

לא ניתן להגדיל יותר את כל החיבורים במידה שווה, מכיוון שישנם שלושה חיבורים המשתמשים בקשת xz (חיבורים 2,3,4), אבל אין רוחב פס פנוי בקשת. לכן סיפקנו ככל האפשר בצורה הוגנת את חיבורים 2,3,4 ונפנה לנסות להמשיך לספק את חיבור 1.

אנו יכולים לספק את חיבור 1 לחלוטין ע"י תוספת 4.67 יחידות רוחב פס:

10/10

3.33/100

**14.67/100**

8/12

6.67/12

**3.33/15**

**11.33/100**

6.67/100

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| חיבור | תחנת מקור | תחנת יעד | מסלול | ר"פ מבוקש | ר"פ זמני |
| 1 | A | C | wy | 8 | **8** |
| 2 | B | C | xzy | 19 | 3.33 |
| 3 | A | D | wxz | 15 | 3.33 |
| 4 | A | D | wxz | 22 | 3.33 |

סוף החישוב.

# שאלה 5

GPS:

נציג את התקדמות הזמן בצורה מפורשת. נזכור כי כאשר חיבור מתחיל או עוצר פעילות, קצבי השליחה בכל החיבורים הפעילים משתנים, לפי יחסי המשקל שלהם.

|  |  |  |  |
| --- | --- | --- | --- |
| t | A | B | C |
| 0 | נוספת חבילה באורך 1 ומתחילה להישלח.  קצב שליחה 0.67 | נוספת חבילה באורך 6 ומתחילה להישלח.  קצב שליחה 0.33 | חיבור לא פעיל |
| 1 | נשלחים 0.67 ביטים. נוספת חבילה באורך 1.  קצב שליחה חדש 0.33 | נשלחים 0.33 ביטים.  קצב שליחה חדש 0.17 | נוספת חבילה באורך 3 ומתחילה להישלח.  קצב שליחה חדש 0.5 |
| 2 | נשלחים 0.33 ביטים.  חבילה נשלחה במלואה. חבילה חדשה מתחילה להישלח | נשלחים 0.17 ביטים. | נשלחים 0.5 ביטים. |
| 3 | נשלחים 0.33 ביטים | נשלחים 0.17 ביטים | נשלחים 0.5 ביטים |
| 4 | נשלחים 0.33 ביטים | נשלחים 0.17 ביטים.  נוספת חבילה באורך 5 | נשלחים 0.5 ביטים |
| 5 | נשלחים 0.33 ביטים.  חבילה נשלחה במלואה | נשלחים 0.17 ביטים. קצב שליחה חדש 0.25 | נשלחים 0.5 ביטים.  קצב שליחה חדש 0.75 |
| *6-ε*  *סטטוס נוכחי* | *החבילה הראשונה התחילה ב-0 וסיימה ב-2.*  *החבילה השנייה התחילה ב-2 וסיימה ב-5. הערוץ סיים פעילות* | *נשלחו עד כה 1 ביט מהחבילה הראשונה,*  *נשארו 5 ביטים + חבילה בגודל 5* | *נשלחו עד כה 2 ביטים מהחבילה, נשאר 1 ביט* |
| 6 | חיבור לא פעיל | נשלחים 0.25 ביטים | נשלחים 0.75 ביטים |
| 6.33 | חיבור לא פעיל | נשלחים 0.083 ביטים. קצב שליחה חדש 1  הסבר חישוב:  R\*Δt=0.25\*0.33=0.085  נשלחו 1.33 ביטים עד כה, נשארו 4.67 ביטים והקצב הוא 1. לכן עוד 4.67 שניות השליחה תסתיים | נשלחים 0.25 ביטים.  חבילה נשלחה במלואה |
| 11 | חיבור לא פעיל | נשלחים 4.67 ביטים.  חבילה נשלחה במלואה. חבילה חדשה מתחילה להישלח | חיבור לא פעיל |
| 16 | חיבור לא פעיל | נשלחים 5 ביטים.  חבילה נשלחה במלואה | חיבור לא פעיל |

החבילה הראשונה של A נשלחה בזמנים 0-2, החבילה השנייה של A נשלחה בזמנים 2-5, החבילה הראשונה של B נשלחה בזמנים 0-11, החבילה השנייה של B נשלחה בזמנים 11-16, החבילה של C נשלחה בזמנים 1-6.33.

WFQ:

מכיוון שכבר סימלצנו GPS ואנו יודעים את זמני הסיום לפי הסימולציה, סדר זמני הסיום הוא סדר שליחת הפקטות ב-WFQ. הסדר הוא A,A',C,B,B' =>

השליחה מתבצעת בצורה בדידה ולא במקביל, לכן החבילה הראשונה של A נשלחה בזמנים 0-1, החבילה השנייה של A נשלחה בזמנים 1-2, החבילה של C נשלחה בזמנים 2-5, החבילה הראשונה של B נשלחה בזמנים 5-11, החבילה השנייה של B נשלחה בזמנים 11-16.

WF2Q:

החבילה הראשונה שתישלח תהיה A, כמו ב-GPS. שליחתה תסתיים ב t=1. כעת אנו צריכים לבחור את החבילה עם זמן הסיום המוקדם ביותר **מבין החבילות הפעילות ב-GPS בזמן t=1.** נבחר את חבילה C. שליחתה תסתיים לאחר 3 שניות, בזמן t=4. החבילה הפעילה בזמן זה ב-GPS עם זמן הסיום המוקדם ביותר היא A', אז נשלח אותה. כעת נשארו שתי החבילות מ-B ונשלח אותן. סה"כ:

החבילה הראשונה של A נשלחה בזמנים 0-1, החבילה של C נשלחה בזמנים 1-4, החבילה השנייה של A נשלחה בזמנים 4-5, החבילה הראשונה של B נשלחה בזמנים 5-11, החבילה השנייה של B נשלחה בזמנים 11-16.

# בונוס חנוכה

שיטות תקשורת בתקופת המכבים:

1. איתותי עשן (ע"י מדורות למשל)
2. שליחים (בתקווה שלא יערפו להם את הראש כשיגיעו אל היעד)