**רשתות תקשורת מחשבים – תרגיל בית 3 חלק תאורטי**

מגיש 1: עמית רוקח

תעודת זהות 1: 322853813

מגיש 2: גיא קוך

תעודת זהות 2: 318962909

**שאלה 1**

ניתן לנצל את שדה הTTL של פקטות כדי לגלות אילו נתבים נמצאים בין המחשב שלנו למחשב אחר בעזרת ICMP, כפי שtraceroute עושה. נניח שהמרחק בין מחשב A למחשב B בנתבים הוא n ונתאר את התהליך המתבצע ממחשב A כדי לגלות אליו נתבים נמצאים בינו לבין מחשב B:

From i=1 to i=n:

1. Computer A sends ICMP packet to computer B with TTL set to i
2. Each router in the path decreases the TTL field by 1 until eventually the i’th router in the path decreases the TTL field to 0.
3. The i’th router discards the packet and sends an ICMP packet that contains it’s IP address with the message “Time Exceeded” back to computer A.
4. A receives the packet sent from the i’th router and prints the IP address of the i’th router (its in the packet).

מאחר שלולאה זו רצה מ1 עד n (המרחק בין מחשב A למחשב B בנתבים) כך מחשב A יגלה את כל הנתבים שנמצאים בינו לבין מחשב B.

**שאלה 2**

א. התפקיד של שורה F הוא שבמידה ופקטה שמגיע לא מתאימה (מבחינת הmask) לתתי הרשתות שבשורות A-E אז היא תתאים לmask שבשורה F, מה שאומר שהיא תפעל בהתאם לשורה F כלומר תצא דרך פורט 3.

ב. להלן טווחי הכתובות בהתאם לmaskים של הרשתות השונות:

* רשת A: 192.168.1.(128-255)
* רשת B: 192.168.(2-3).(0-255)
* רשת C: 192.168.1.(128-191)
* רשת D: 192.168.1.(160-191)
* רשת E: 192.168.3.(128-255)
* רשת F: (0-255).(0-255).(0-255).(0-255)

על בסיס הmaskים וLongest Prefix Match ננתב את החבילות:

* P ינותב ליציאה 3 (כי הLongest Prefix Match שלו הוא שורה F)
* Q ינותב ליציאה 1 (כי הLongest Prefix Match שלו הוא שורה A)
* R ינותב ליציאה 2 (כי הLongest Prefix Match שלו הוא שורה B)
* S ינותב ליציאה 2 (כי הLongest Prefix Match שלו הוא שורה B)
* T ינותב ליציאה 2 (כי הLongest Prefix Match שלו הוא שורה E)

ג. אכן, ניתן למחוק את שורה E.  
הסבר: בהתאם לMask של שורה E וכך שאין שורה אחרת שהmask שלה היא תת רשת בMask של E נובע כי הIP של של כל פקטה שמנותבת לפי שורה E הוא בפורמט 192.168.3.(128-255). הMask של שורה E הוא תת רשת בMask של שורה B שהוא 192.168.(2-3).(0-255) וגם בmask של שורה F, אך, כל פקטה שמתאימה לMask של E תהיה עם longest prefix שמתאים לMask של שורה B, ולכן, כל פקטה שמתאימה לשורה E תנותב בהתאם לשורה B. כלומר, כעת כל פקטה שקודם לכן נותבה לפי שורה E תצא מפורט 2 (פורט היציאה של שורה B), פורט זהה לפורט היציאה של שורה E. בנוסף, פקטות שקודם לכן לא נותבו בהתאם לשורה E, ינותבו באותו אופן בו נותבו קודם לכן. מכאן שבמחיקת שורה E, מחקנו שורה מהטבלה מבלי לשנות את הניתוב של אף הודעה.

**שאלה 3**

א. טבלת הניתוב בצומת A בתחילת התהליך:

|  |  |  |
| --- | --- | --- |
| **Cost** | **Next Hop** | **Destination** |
| 0 | - | **A** |
| 4 | B | **B** |
| Inifinity | - | **C** |
| Inifinity | - | **D** |
| Inifinity | - | **E** |
| 1 | F | **F** |

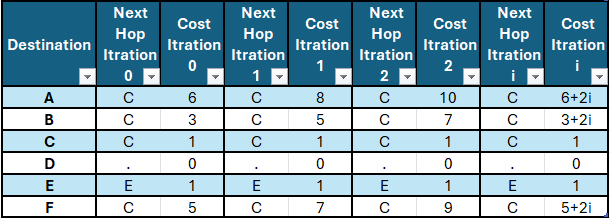
טבלת הניתוב בצומת A בסיום התהליך:

|  |  |  |
| --- | --- | --- |
| **Cost** | **Next Hop** | **Destination** |
| 0 | - | **A** |
| 3 | F | **B** |
| 4 | F | **C** |
| 3 | F | **D** |
| 2 | F | **E** |
| 1 | F | **F** |

ב. האלגוריתם יתייצב מאחר שיש מסלול בין כל שתי צמתים.

|  |  |  |
| --- | --- | --- |
| **Cost** | **Next Hop** | **Destination** |
| 0 | - | **A** |
| 3 | F | **B** |
| 5 | F | **C** |
| 6 | F | **D** |
| 7 | F | **E** |
| 1 | F | **F** |

ג. האלגוריתם לא יתייצב מאחר שיש לפחות זוג צמתים אחד שאין מסלול ביניהם (למשל B וD), ולכן, אין מגבלה על המרחקים שצומת C וצומת D "יציעו" זה לזה במסגרת העדכונים ונקבל מקרה של count to infinity.

טבלת המרחקים של צומת D לאורך זמן תראה באופן הבא:  


כשi מייצג מספר איטרציה שהולך לאינסוף.