

# פרויקט, 3

שחר פרץ

18 בנובמבר 2024

## JAVA STUFF ..... (1)

אני לא אסכם הכל כי לא נראה לי חשוב. שאלה: מה ההבדל בצריכת הזכרון בין שתי הבאות:

```
1 float[] = new float[10]
2 var A = new ArrayList<Float>
```

' נתבונן בשני דברים: float[10] יהיה 40bytes  $\rightarrow 10 * 32\text{bit}$  בעוד ArrayList, תופס בלוק בזכרון שמכיל array של רפרנסים לאובייקטי Float. בדיפולט, הוא ייצור מערך של 6 איברים.

### 1.1 Bytes and Characters

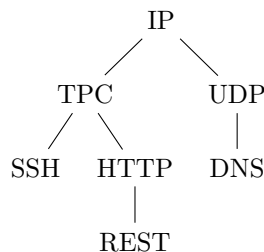
סטרינג הוא אוסף של תווים. ב-byte אחד אפשר להעביר תו ASCII אחד. UTF-8 הוא קידוד תואם unicode ותואם ascci (כלומר קידוד utf-8 יהיה זהה לקידוד ASCII בהינתן אותם הספרות). בשביל לקרוא utf-8 צריך להבין אם התו הבא הוא חלק מהתו או לא. במקרה של עברית, כל תו הוא שני ביטים.

java מחזיק בזכרון בקידוד utf-16. זה למעשה מערך של shortint.

הערה: byte shifting היא הפכולה שמתייחסת להעברת ביטים זכרון. דוגמה:  $3 \gg 3$  &  $15 \gg 2$ . המרה ב- & FFFFFFF יוצרת מפעילה or byte0sidr. בשביל אפרטורים כמו 4 & 7 < 2 צריך לבדוק precedence.

## TCP/IP ..... (2)

הרשת שרוב האינטרנט עובד איתה היא רשת IP, כאשר לכל node (מחשב) יש כתובת IP. דוגמאות לכתובות ipv4 הן 10.0.0.1 '132.52.32.20. גם כתובות ipv6 ארוכות יותר נפוצו, אך לא באותה המדיה. יש לכל מחשב כמה כתובות ip, הן 127.0.0.1 (localhost, משתמש לתקשורת עצמאית), חיבור ל-wifi (לדוגמה 10.0.0.5) וחיבור לקווי (192.8.0.3). מעל IP קיימים פרוטוקולים נוספים, כמו TCP ו-UDP. מעל TCP נמצאים פרוטוקולים כמו http, ssh, ועוד. פרוטוקולים כמו DHCP מקצים כתובות IP ע"י ה-router. על ה-UDP רץ DNS.



האימייל האישי: tal.franjigmail.com.

איך ניגשים לאינטרנט? יש לנו client (אנחנו) המתחברים ל-server (נגיד, cnn.com). מעשית, cnn בנוי מ-node balancer, שמאחוריו יש מאות מחשבים הקרויים front-end servers (הם מייצרים HTML וכו') ומתקשרים עם backend servers שמבצעים פעולות ספציפיות. השפה הכי נפוצה ל-backend היא java. ב-FE יש HTML, react, node.js וכו'.

ה-localhost נמצא רק על המחשב, יש רשת פרטית בין הראוטר, ורשת עולמית. כל הרשתות נפרדות וכתובות חח"ע ועל באותה הרשת.

אצלנו אין BE. כל הרשת היא מבוזרת, וכל שרת מבצע את כל העבודה. התקשורת בין המחשבים ב-TPC.

כאשר שני clients על אותה הרשת המקומית (או יותר), נתקל בבעיה - כאשר client אחד יבקש להתחבר למחשב אחר, לא ברור לאיזה client ירצה להתחבר. לכן, ב-TCP צריך לספק לא רק IP אלא גם port number, כאשר כל client חושף את עצמו דרך port. לדוגמה, port 80 הוא ה-default ל-browser.

כאשר תרצה להיות שרת, תרצה לבצע דבר שקוראים לו האזנה (listening). תתבצע פעולה שקוראים לה accept, בעבור port מסויים. היא מאפשרת לקבל מידע מפורט. היא מחזירה דבר הקרוי socket. אם נרצה לקבל סוקטים נוספים, נצטרך להריץ את כל הסיפור בלולאה. התקשורת דרך ה-socket מתבצעת ע"י מעבר ביטים. צריך להיות מסוגלים לטפל ביותר מ-socket אחד, לדוגמה כאשר שני מחשבים שולחים

לנו blockchain. אחת מדרכים לעשות את זה, היא Threads.

רק שרת אחד יכול להשתלט על פורט נתון. אם ה-process חיי וקיים, צריך להרוג אותו. יכול להיות גם זומבי זמנית מערכת ההפעלה חיי ותופס את הפורט אבל לא באמת עושה דברים. הפורטים מתחת ל-256 שמורים למערכת וצריך הרשאות root בשביל לגשת אליהם. נתבונן בדוגמה של TrivialServer.java בשביל להבין איך הדברים האלו עובדים. הבעיה במודל שנמצא שם (טיפול בכמה clientes באמצעות לולאה) יכול לארוך הרבה זמן, כך ש-clients אחרים לא יכלו להתחבר. ה-browser הוא HTTP, כלומר, הוא קורה הודעות שממש מופרדות ב-newlines, ולכן נצטרך להפוך את ה-input מה-socet להודעה. הקוד שם כתוב עם באגים כי הוא קרוא רק את 1024 הביטים הראשונים ולא ייתחס ליותר מזה. נעשה parsing ל-buffer' כלומר נמיר את הביטים לערימת טקסט ונפרמט אותה בצורה קריאה. נצטרך לספק תשובה בפרוטוקול HTTP. אם נוסף koko/ לסוף ה-url, אז זה ישלח כחלק מבקשת ה-HTTP. שימו לב – זה שרת אידיאלי ולא יעבוד בדברים קצת יותר גדולים.

בהתחלה השרת שהמורה יעלה כנראה יהיה הכי יציב וכולנו תחבר אליו. השרת מתישהו ירד, המטרה היא שהרשת תהיה מבוזרת ותחזיק את עצמה.

## 2.1 ניתוב ברשת

תהי רשת מחשבים עם צורת חיבור ("טופולוגיה"). יהיה גרף:

$$a \rightarrow \{b, e, d\}, c \rightarrow b$$

דרך לתקשר בין node-ים היא באמצעות טבלאות של node-ים וטבלאות ניתוב שכל שרת יעביר פעם בכמה זמן ל-node-ים האחרים, ליידע מי מחובר אל מי.

## 2.2 port-forwarding

יהיה מחשב בבית וראוטר של בזק או לא של בזק. כאשר נצא מהרשת הלוקאלית בין המחשב לראוטר, לאינטרנט (בה"כ ל-cnn.com), הראוטר יאסור על דברים מהאינטרנט להיכנס לרשת הביתית מסיבות של אבטחה. הדרך להעביר מידע, היא תהיה שהשרת מבחוץ יתחבר ל-port של ה-router ויעביר אותו ל-port של המחשב. צריך לקנפג את זה ידנית בהגדרות ה-router. המשימה לשיעור הבא – לנסות שחבר יתחבר מהבית שלו ל-TrivialServer שנריץ על המחשב שלנו.