



1. קראו את סיכום הרצאה 1 + 2. מהי שפה אוטרקית?

שפה אוטרקית היא שפה אשר יש קטע קוד יחיד המגדיר את התוכנית. לדוגמה בשפת פסקל קטע הקוד יתחיל בprogram ויסתיים ב-end. בפרט בשפה אוטרקית ניתן לעשות שימוש במילים שמורות ובמילים שהוגדרו באותו קטע קוד, לדוגמה בפסקל, write, writeln, הן מילים שמורות. בניגוד לכך, בשפה מטאפיזית ניתן לקשר קטעי קוד ממקומות שונים, לדוגמה בשפת C יצירת קלט ופלט הינן שגרות הממומשות בקטע קוד אחר.

2. קראו את הסיכום בנושא שמות, קישורים, ויִשְׁיוֹת בְּנוֹת-שֵׁיִם. מה ההגדרה של המונח nameable? מה התרגום לעברית או לערבית או לרוסית של המונח nameable?

Nameable בעברית – בר שיום – דבר מה שאפשר לתת לו שם. מילים ברות שיום הינן מילים חוקיות בשפה ( שלרוב מוגדרות על ידי ביטוי רגולרי ) למעט מילים שמורות אשר פעולות פעולות בשפה והמתכנת לא יכול לתת להן משמעות אחרת.

3. קראו את המכתב למערכת העיתון Communications of the ACM שכתב Donald Knuth בשנת 1964 בנושא Backus Normal Form vs. Backus Naur Form. מה ההבדל, אם יש הבדל, בין צורה קנונית (canonical form) וצורה נורמלית (normal form)? מה ההבדל, אם יש הבדל, בין Backus Naur Form ובין Backus Normal Form?

ההבדל המהותי בין הצורה הקנונית לבין הצורה הנורמלית היא באופן השוואת האובייקטים. בצורה נורמלית שתי אובייקטים הם שווים אם תכולתם שווה לעומת הצורה הקנונית בה שתי אובייקטים שווים הן אם תכולת האובייקטים שווה והן הסדר של השדות באובייקט זהה. לדוגמה אובייקט המכיל שתי שדות זהים אך בסדר שונה יהיו זהים נורמלית אך לא קנונית. על פי המכתב אין הבדל משמעותי בין Backus Naur Form ובין Backus Normal Form, אך השינוי בשמות נובע מרצון לתת הכרה לNaur, ראשי תיבות BNF, Form אינו Normal.

4. ציינו כמה שגיאות בשקפים של פרק 2.4, אם ישנן כאלה, והציעו תיקון. לפי ההסבר של פ"ר לורנז בהרצאה, בשקף 8 נמצאת הטעות, שכן אם  $e$  שייכת לשפה היא יכולה להיות רצף של תווים ולכן נצפה שה- $*$  תהיה מחוץ לסוגריים בכדי להכליל את הרצף כולו.

Kleene closure

$$e \in RE(\Sigma) \Rightarrow (e^*) \in RE(\Sigma)$$

שקף 9 ממחיש שגיאה זאת שכן כעת  $e'$  הינה מילה אחרת ושם אכן הכוכבית נמצאת מחוץ לסוגריים.

• Kleene closure:

$$e = (e')^* \Rightarrow S(e) = \bigcup_{i=0}^{\infty} S(\overbrace{e' \dots e'}^{i \text{ times}})$$

5. מה מציין הסמל  $\varepsilon$ ? ציינו שלוש משמעויות בשלושה הקשרים שונים.

במדעי המחשב – מסמל מילה ריקה בביטויים רגולריים.  
בחשבון אינפיניטסימאלי - מספר מאוד קטן.  
בתורת החשמל – מייצג מקדם דיאלקטרי.

6. מנו את כל האפילויות שנהוגות בבנאי הטיפוסים בפסקל.

- בנאי של מערך - לא ניתן ליצור מערך של פונקציות
- בנאי של SET - לא ניתן ליצור סט של סטים
- בנאי של קובץ - לא ניתן ליצור קובץ של פונקציות
- בנאי של RECORD - לא ניתן ליצור RECORD של פונקציות

7. מהם ה-nameables של שפת BATCH של DOS?

- שמות של משתנים.
- תגיות עבור קפיצות מסוג goto

8. כתבו או מצאו באינטרנט תוכנית בפסקל שמדפיסה את עצמה. הסבירו את התוכנית.  
מה ה"טריק" שמאפשר זאת?

```
const a='const a=';b=';begin write(a,#39,a,#39#59#98#61#39,b,
#39,b)end.';begin write(a,#39,a,#39#59#98#61#39,b,#39,b)end.
```

הטריק הוא לכתוב את כל התכנית בשורה אחת, את התווים החסרים מוסיפים בקוד האסקי שלהם ולכן לא צריך להוסיף פסיק ביניהם. התכנית נלקחה מ-  
[https://www.nyx.net/~gthompso/self\\_pasc.txt](https://www.nyx.net/~gthompso/self_pasc.txt)

```
C:\Users\kwarta\Documents\Technion\winter18\ProgrammingLanguages\hw2>fpc quine.pas
Free Pascal Compiler version 2.6.4 [2014/03/06] for i386
Copyright (c) 1993-2014 by Florian Klaempfl and others
Target OS: Win32 for i386
Compiling quine.pas
Linking quine.exe
0 lines compiled, 0.1 sec , 25856 bytes code, 1692 bytes data

C:\Users\kwarta\Documents\Technion\winter18\ProgrammingLanguages\hw2>quine.exe
const a='const a=';b=';begin write(a,#39,a,#39#59#98#61#39,b,#39,b)end.';begin write(a,#39,a,#39#59#98#61#39,b,#39,b)end.
C:\Users\kwarta\Documents\Technion\winter18\ProgrammingLanguages\hw2>
```

9. עיינו בתיעוד שפת התכנות **Kotlin**:

a. הסבירו את המושגים frontend ו-backend של מהדר (קומפיילר)? מנו 3 backend-ים של המהדר של kotlin.

פרונט-אנד מתעסק בשפה עצה ואילו בק-אנד בשפת המכונה.  
פרונט אנד מבצע parsing לקוד ובניית עץ התרגום. לאחר מכן  
הבק-אנד לוקח את הפלט של הפרונט-אנד ובונה ממנו קוד מכונה

בק-אנד ספציפית עבור kotlin:

- java script
- android apps
- server side apps

b. באילו מנגנוני הגדרות של שפות תכנות משתמשת Kotlin, ובאיזה אופן?

שפת Kotlin דומה בתכונותיה לשפת Scala, הבדל מהותי בין השפות הוא זמן הקמפול המהיר של Kotlin ביחס לשפת Scala. אחת ממטרות כותבי השפה הוא זמן קמפול מהיר של תוכניות כמעט כמו JAVA. ניתן לכתוב בה פקודות פרוצדורליות בדומה ל-C++ (בניגוד ל-JAVA בו לא ניתן להגדיר פונקציה מחוץ למחלקה). בדומה לשפת APACHE GROOVY ו-SCALA אין חובה בסיום פקודה ב-";" אלא מספיק לרדת שורה. כלומר סיום פקודה בנקודה פסיק הוא אופציונאלי.

c. האם השפה אוטרקית? באיזה אופן נקבעים גבולות התכנית? באיזה אופן נקבעת נקודת תחילת הביצוע של התכנית?

- השפה אינה אוטרקית מכיוון שניתן לבצע import
- גבולות התכנית נקבעות ע"י גבולות התכנית של כל התכניות אשר מיובאות לתכנית הנתונה בזמן ריצה ע"י import
- נקודת תחילת הביצוע נקבעת ע"י הפונקציה main

d. מהם סוגי הישויות להם ניתן לתת שמות (nameable) בשפה?

- מחלקות, namespaces, פונקציות, משתנים ותוויות לחזרה מפונקציות

e. האם אפשר להשתמש בשפה באותו שם לישויות מסוגים שונים? אם כן, תנו דוגמה.

- כן ניתן להגדיר משתנה פנימי בפונקציה בעל שם זהה לשם הפונקציה. הדפסת ערך הנ"ל ידפיס את ערך **המשתנה** (ולא הפונקציה)

f. מהם בנאי הטיפוסים בשפה?

- מכיוון שהשפה הינה שפה מונחית עצמים אזי כל טיפוס בה מיוצג ע"י אובייקט ולכן הבנאים היחידים הינם בנאי מחלקות