

חלק 6 : תיאורטי

קוט אנטולי – 324413756

(9)

א) במערכת אובייקטים שבנינו בכיתה (של **Shmython**) לא ניתן להוסיף פונקציות למאפיינים של מופע (**instance attributes**)

לא נכון , כן ניתן להוסיף למילון של המופע להלן הדוגמא :

```
t['set']('fun', lambda : print("function"))  
t['get']('fun')()
```

ב) במפרש של מחשבון (**calculator**) שכתבנו בכיתה פונקציות **calc_eval** ו-**calc_apply** עובדות בצורת רקורסיה הדדית.

לא נכון , רקורסיה הדדית זה כאשר פונקציה רקורסיבית קוראת לפונקציה רקורסיבית אחרת שקוראת שוב לראשונה , במפרש של המחשבון רק הפונקציות **Analyze** ו **analyze_operands** קוראות אחת לשנייה

ג) מטרת **memoization** היא לייעל חישוב רקורסיבי ע"י הקטנת ניצול זיכרון על חשבון זמן ריצה.

נכון , כמו בדוגמא שעשינו בכיתה עם **fibunachi** , יצרנו מילון ששומר ערכים קודמים של הפונקציה הרקורסיבית ולכן לא מבצע אותם שוב וחוסך בזמן ריצה

ד) ב- **Python 3.x** משתמשים בהצהרה **nonlocal** על מנת לעדכן תוכן של מבנה נתונים (למשל מילון) בסביבה כוללת (לא כולל מסגרת גלובאלית).

לא נכון , משתמשים ב **NONLOCAL** כדי לעדכן טיפוסים שלא ניתנים לשינוי כמו **int** , **float** מילון הוא טיפוס שכן ניתן לשינוי לכן הוא לא זקוק ל **NONLOCAL**

ה) במפרש של מחשבון (**calculator**) שכתבנו בכיתה פונקציית **calc_eval** מממשת **רקורסית עץ** על מנת להעריך את הארגומנטים של הביטוי המוערך.

נכון ,

ו) **פולימורפיזם בהורשה** ממומש בעזרת ממשק משותף בין הטיפוסים.

נכון , בהורשה פשוט דורסים פונקציות של האבא , כמו **STR** או **REPR**

ז) פונקציות גנריות (**generic functions**) המוגדרות על פרמטרים מטיפוסים זרים (**parametric polymorphism**) ממומשות ע"י הגדרת ממשק משותף בין הטיפוסים של ארגומנטים.

נכון

