



תור כמבנה נתונים חלק ב'

תרגול



תרגול היסודות

תרגיל זה זהה לתרגיל המופיע ביחידת התרגול בחלק א', אך כאן אסור לכם להשתמש בספרייה המובנית `queue` (תור).

1. כתבו תוכנית בפייתון שיוצרת תור חדש בשם `q`
2. הוסיפו לתור `q` את המחרוזת `"(-:"`
3. הוסיפו לתור `q` את המספר 2
4. הוציאו את האיבר שנמצא בראש התור והדפיסו אותו
5. השתמשו בפעולה שבודקת אם התור ריק או לא, והדפיסו את התוצאה
5. שוב, הוציאו את האיבר שנמצא בראש התור והדפיסו אותו
6. שוב, השתמשו בפעולה שבודקת אם התור ריק או לא, והדפיסו את התוצאה - מה קיבלתם הפעם?

מימוש חלופי

מורה למחשבים הציג לתלמידים בכיתה את הפעולה `get` שמוציאה איבר מראש התור.

```
def get(queue):
    if empty(queue):
        return ("תור ריק")
    else:
        return queue.pop()
```

תלמיד בשם עידן טען שאפשר לממש את הפעולה בצורה אחרת כך שהיא עדיין תעבוד אותו הדבר. הוא ניגש ללוח וכתב את המימוש הבא (השינוי מודגש בכתום):

```
def get(queue):
    if empty(queue):
        return ("תור ריק")
    else:
        return queue[0]
```

המורה ענה לו בתגובה שהמימוש הזה לא נכון. האם תוכלו להסביר למה?

קלט מהמשתמש

כתבו תוכנית שקולטת מספר מהמשתמש, ומבצעת את הפעולות הבאות בהתאם למספר שהוקש:

- אם המשתמש הקיש 1 - התוכנית תדפיס את מספר האיברים שיש בתור.
- אם המשתמש הקיש 2 - התוכנית תדפיס את כל האיברים בתור.
- אם המשתמש הקיש 3 - התוכנית תדפיס את כל האיברים בתור בסדר הפוך.
- אם המשתמש הקיש 4 - התוכנית תדפיס את המספר עם הערך המקסימלי בתור.
- אם המשתמש הקיש 5 - התוכנית תדפיס את המספר עם הערך המינימלי בתור.

הנחיות: כדי לבדוק את התוכנית שלכם, צרו תור עם מספרים בלבד. צרו פעולות עזר במידת הצורך.

סקרנות

כתבו פעולה בשם `peek_last` שמקבלת תור כפרמטר.

```
def peek_last(queue):
    <כתבו כאן את הקוד שלכם>
```

הפעולה תאפשר "להציץ" אל האיבר **בסוף** בתור. כלומר, היא תחזיר לנו את האיבר בסוף התור ועדיין תשאיר אותו באותו מקום בתור מבלי להוציא אותו.

השתמשו בפעולה בתוכנית הראשית שלכם כדי לוודא שהיא אכן עובדת.

פולינדרום

פולינדרום היא מילה, מספר, משפט או כל רצף סמלים אחר שקריאתו מימין ומשמאל היא זהה (לדוגמה: שמש, לבלבל)

כתבו פעולה בשם `is_palindrom` שמקבלת תור שהאיברים שלו הם מחרוזות.

```
def is_palindrom(queue):  
    <כתבו כאן את הקוד שלכם>
```

הפעולה תדפיס "yay" אם האיברים בתור יוצרים פולינדרום, אחרת תדפיס "nay". לדוגמה:

```
def main():  
    q = new_queue()  
    put(q, "A")  
    put(q, "B")  
    put(q, "B")  
    put(q, "A")  
    is_palindrom(q)
```

התוכנית תדפיס yay.

ועבור התוכנית הבאה:

```
def main():  
    q = new_queue()  
    put(q, "A")  
    put(q, "B")  
    put(q, "B")  
    put(q, "A")  
    put(q, "C")  
    is_palindrom(q)
```

התוכנית תדפיס nay.

הנחיה: באפשרותכם להוציא איברים מן התור כרצונכם.