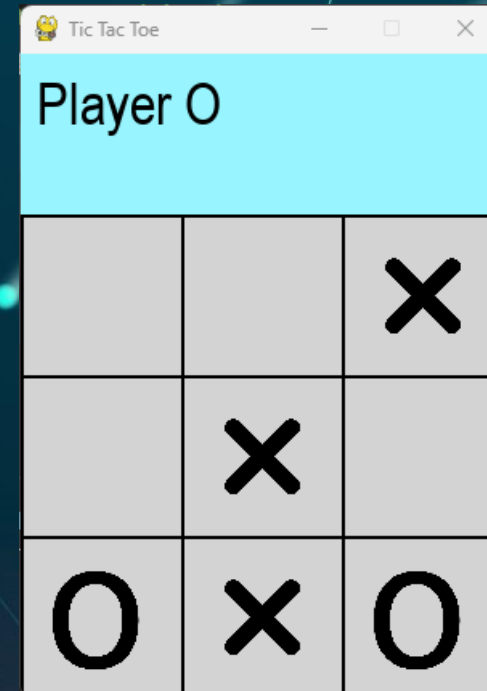




קריית החינוך  
פארק המדע  
בית לערכים  
למצוינות ולחדשנות

# משחק איקס עיגול Monte Carlo Method



# דוגמה – מונטה קרלו במשחק איקס עיגול

- נדגים כיצד ליישם את אלגוריתם מונטה קרלו במשחק איקס עיגול.
- כיוון במשחק איקס עיגול יש לכל היותר  $3^9 = 19,683$  מצבים (ויש אפילו פחות), ניתן ליישם את המשחק באמצעות תכנון דינמי.
- נאמן את המודל שלנו מול סוכן רנדומלי. הסוכן שלנו ישחק את השחקן הראשון עם איקס.

## • תרגיל

- עליכם לשנות את הקוד ולבנות סוכן שישחק את שחקן 2 – המשחק עם העיגול.

# מימוש Q-Table

- אחת הדרכים לממש Q-Table היא באמצעות מילון (Dictionary):
- המפתח (key) יהיה (state, action)
- הערך (Value) יהיה הערך המספרי של המצב-פעולה.

• נסביר בקצרה כיצד ממומש מילון בפייתון, וכיצד נוכל לבנות מילון שהמפתחות שלו הם (state, action).

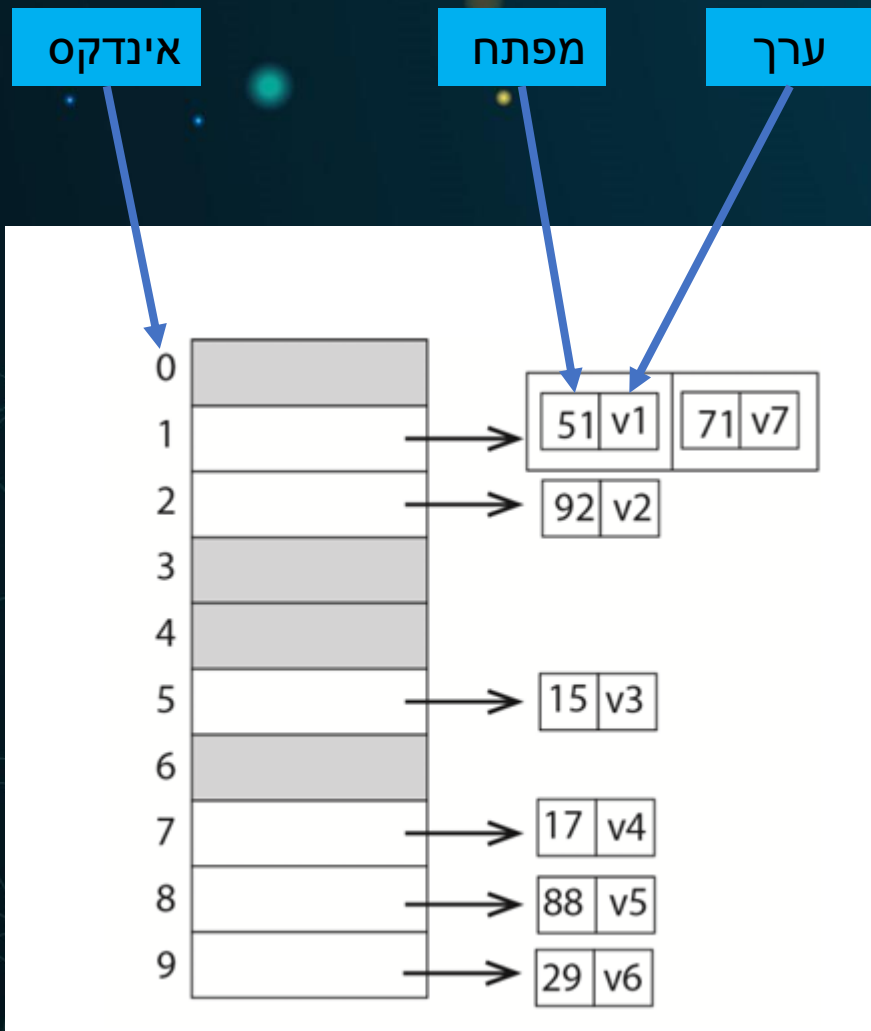
• תזכורת לפייתון:

$$Q[(state, action)] = value$$

- אם המפתח קיים במילון – הערך יעודכן. אין כפילויות.
- אם המפתח לא קיים במילון – המפתח יתווסף ויקבל את הערך הנתון.



# טבלת גיבוב - Hash Table



- מילון בפייתון ממומש על ידי Hash Table (וכך גם set).
- טבלת גיבוב בנויה ממערך הכולל "מפתח" שהוא האינדקס, וערכים.
- בנוסף מוגדרת פונקציית hash הממירה את הערכים למפתח.  $h(\text{key}) = \text{index}$ .
- לדוגמה, במקרה זה הפונקציה היא modulo 10, כך שהתוצאה היא תמיד בין 0-9.
- הכנסת איבר נעשית על ידי חישוב/מיפוי המפתח (mapping) והכנסתו למקום המתאים.
- אם המקום תפוס ("התנגשות") אז אחד הפתרונות הוא להשתמש בתור ולשרשר את הערכים.

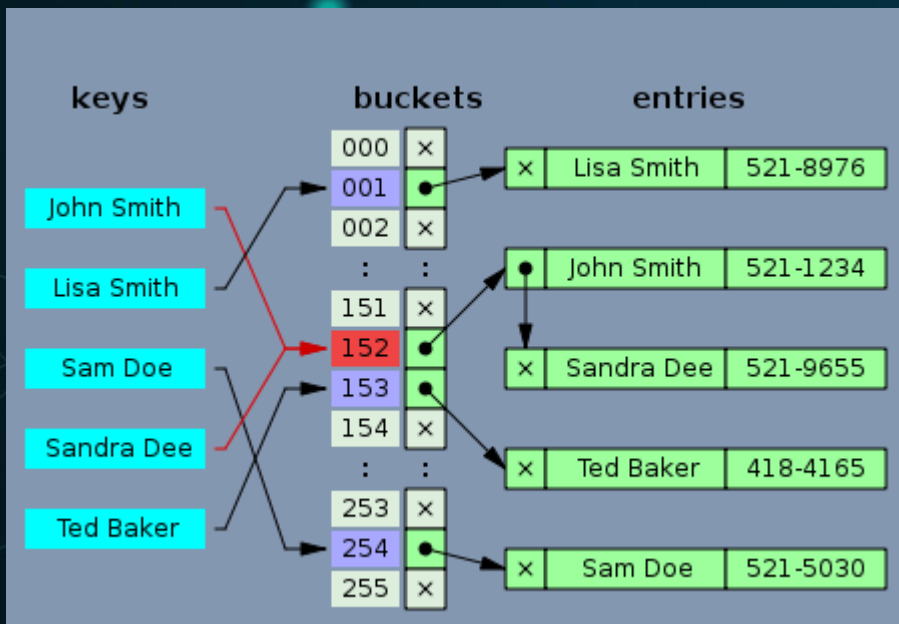


# פונקציית גיבוב

- ניתן להשתמש בכל סוגי ערכים באמצעות פונקציית hash מתאימה.

- למשל, שימוש בפונקציה הממירה מחרוזת למספר בין 0-256 יאפשר לנו לבנות טבלת hash שהמפתחות הם מחרוזות והערכים מספרי טלפון.

- סיבוכיות זמן תלויה בגודל המערך (כמה מפתחות ריקים), ובהתפלגות של פונקציית הגיבוב (כמה התנגשויות יש).

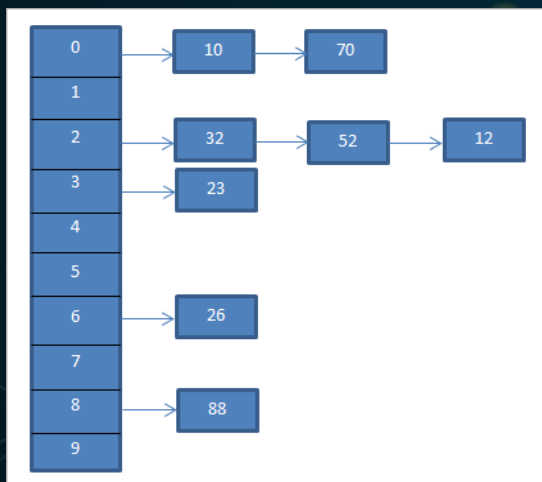


במקרה הגרוע	ממוצע	סיבוכיות זמן
$O(n)$	$O(1)$	חיפוש
$O(n)$	$O(1)$	הכנסה
$O(n)$	$O(1)$	הוצאה



קריית חינוך "פארק המדע"  
בית לערכים, למצוינות וחדשנות

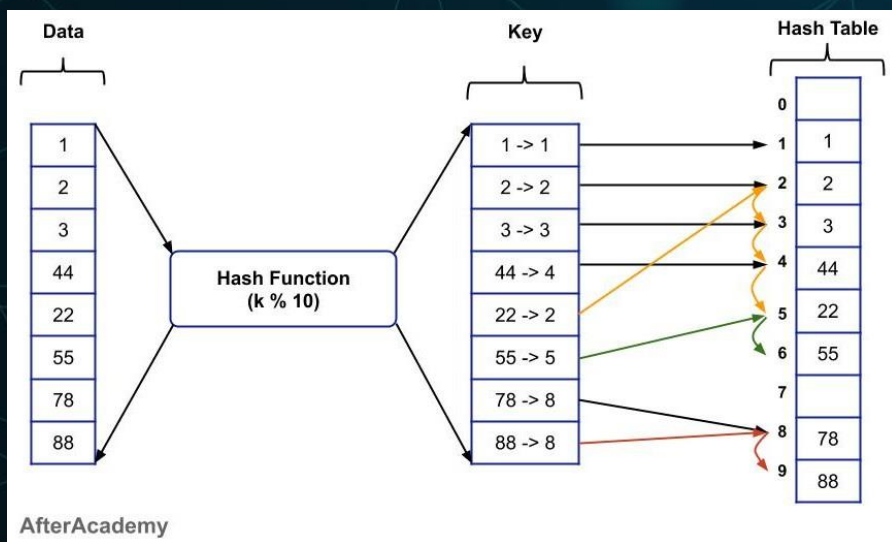
# טיפול בהתנגשויות Hash Table



נציג שתי שיטות לטיפול בהתנגשויות:

- טבלה סגורה – במקרה של התנגשות נשמור, למשל, תור של ערכים עם אותו מפתח.

- טבלה פתוחה – במקרה של התנגשות מחפשים מקום חדש בטבלה לפי כלל קבוע מראש:



- למשל המקום הבא בטבלה.

- למשל הפעלה חוזרת של פונקציית הגיבוב ומציאת המקום הבא.



קריית חיבור "פארק המדע"

בית לערכים, למצוינות וחדשנות

# פונקציית גיבוב לאובייקטים

- פונקציית גיבוב מקבלת ערך ומחזירה מספר שלם (האינדקס של הטבלה).
- על מנת שערך כלשהו יוכל לשמש כמפתח נדרש להגדיר עבורו פונקציית גיבוב.
- בפייתון לכל המשתנים הבסיסיים (מספרים, מחרוזות) יש פונקציית גיבוב מוכנה, ולכן ניתן להשתמש בהם כמפתחות במילון.
- לאובייקטים אחרים עלינו להגדיר פונקציית גיבוב על מנת להשתמש בהם כמפתחות במילון.
- אנחנו נוסיף למחלקה State הגדרה של פונקציית גיבוב המסתמכת על הלוח, כך:

```
def __hash__(self) -> int:  
    return hash(repr(self.board))
```