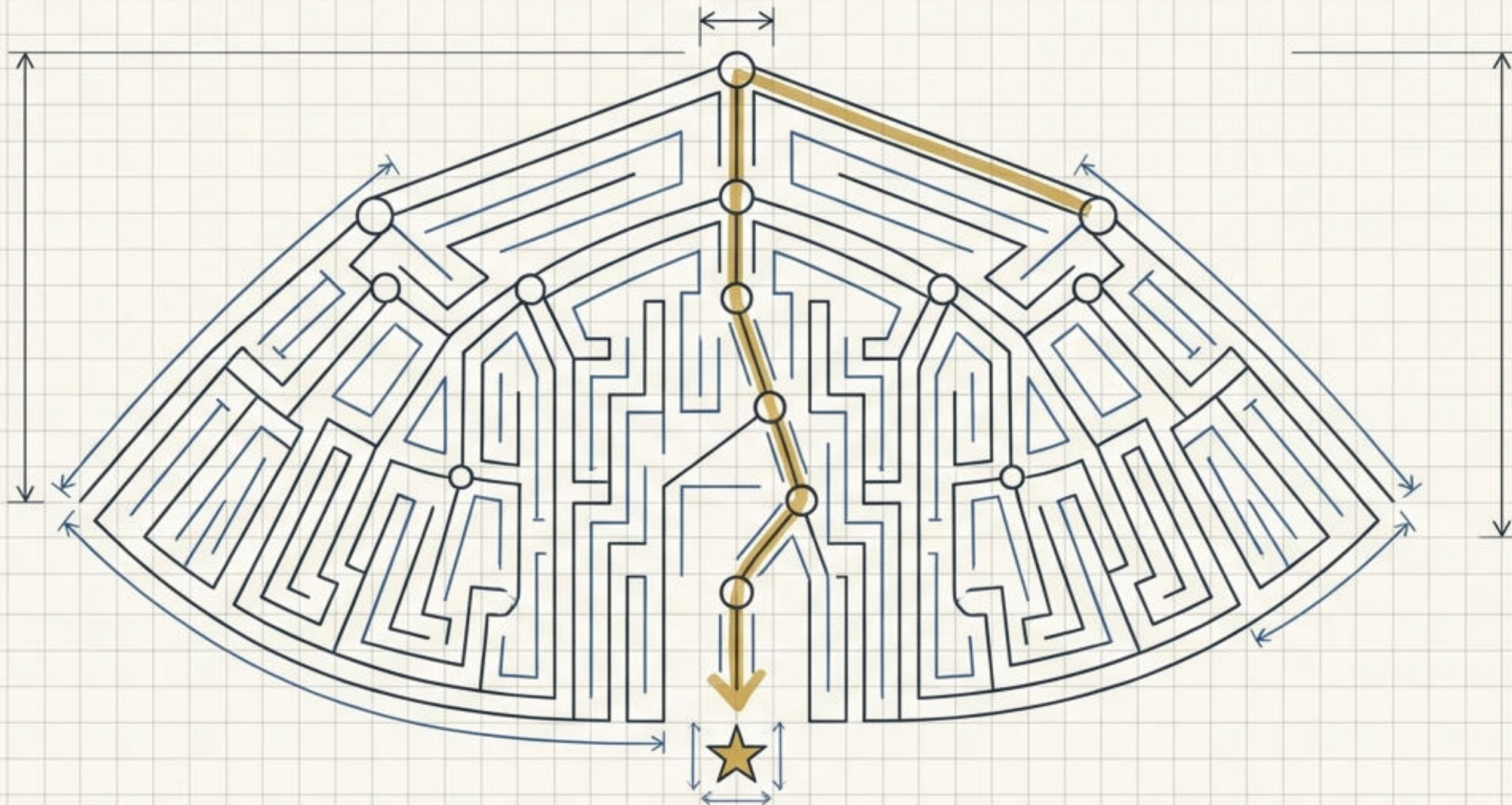




מבוא לבינה מלאכותית: מסוכנים פשוטים לחיפוש חכם

מדריך חזותי לפתרון בעיות באמצעות אלגוריתמי חיפוש – מ-BFS ועד A^*

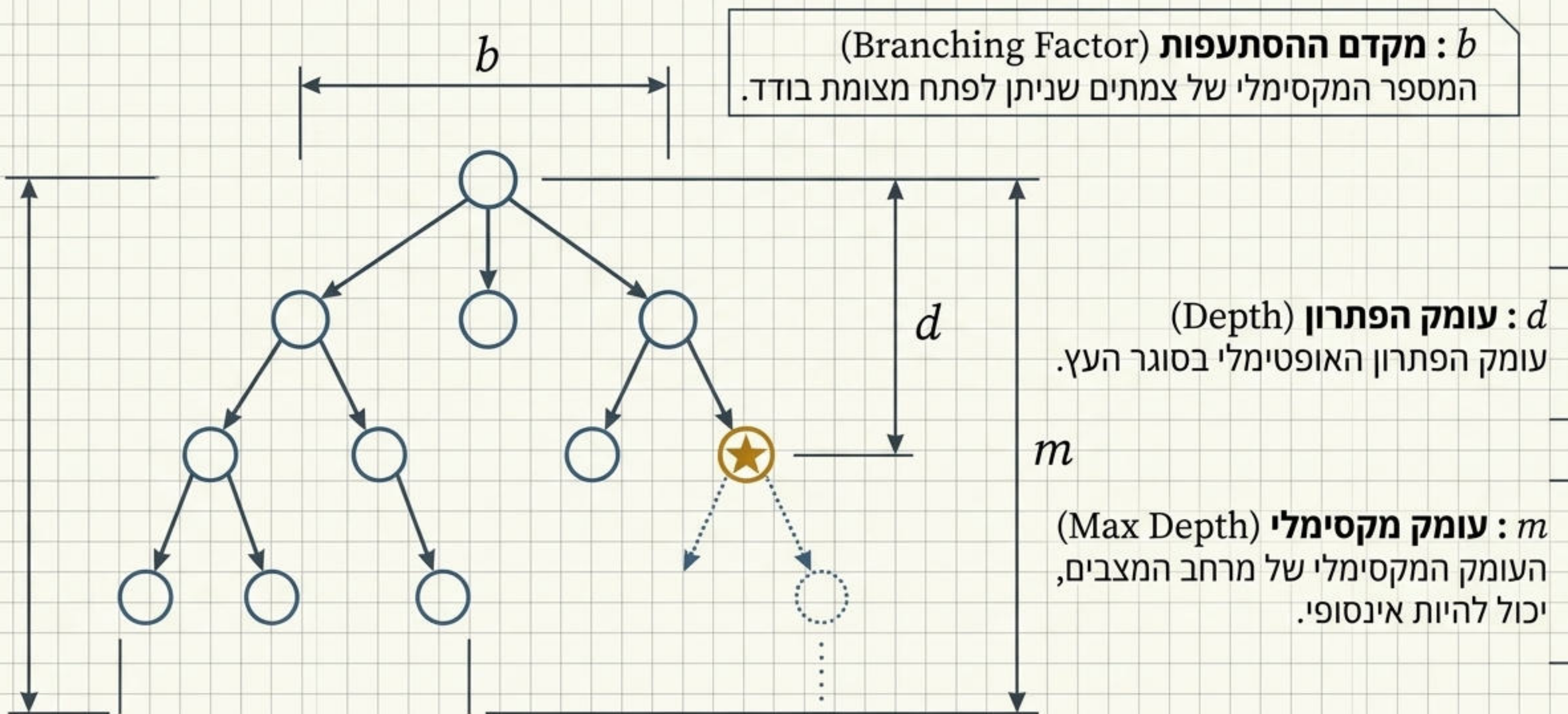
	אנושי (Human-like)	רציונלי (Rational)
חשיבה (Thinking)	מערכות שחושבות כמו בני אדם (מודלים קוגניטיביים)	מערכות שחושבות בצורה רציונלית (לוגיקה טהורה)
פעולה (Acting)	מערכות שפועלות כמו בני אדם (מבחן טיורינג)	מערכות שפועלות בצורה רציונלית (סוכנים רציונליים)

המוקד שלנו בקורס: **בניית סוכנים רציונליים** (Rational Agents) – מערכות שמקבלות את ההחלטה הטובה ביותר האפשרית כדי להשיג מטרה מוגדרת.



כדי לבחור את הפעולה הנכונה, הסוכן מתכנן קדימה באמצעווש במרחב מצבים.

האנטומיה של עץ חיפוש: שפת המדדים



4 מדדי ההערכה: איך שופטים אלגוריתם?



שלמות (Completeness)

האם האלגוריתם
תמיד ימצא פתרון
אם קיים כזה?



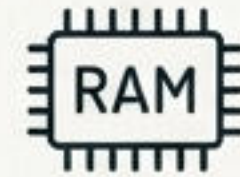
אופטימליות (Optimality)

האם האלגוריתם
תמיד ימצא את
הפתרון הזול או
הקצר ביותר?



זמן (Time Complexity)

כמה צמתים
האלגוריתם צריך
לייצר או לפתח?

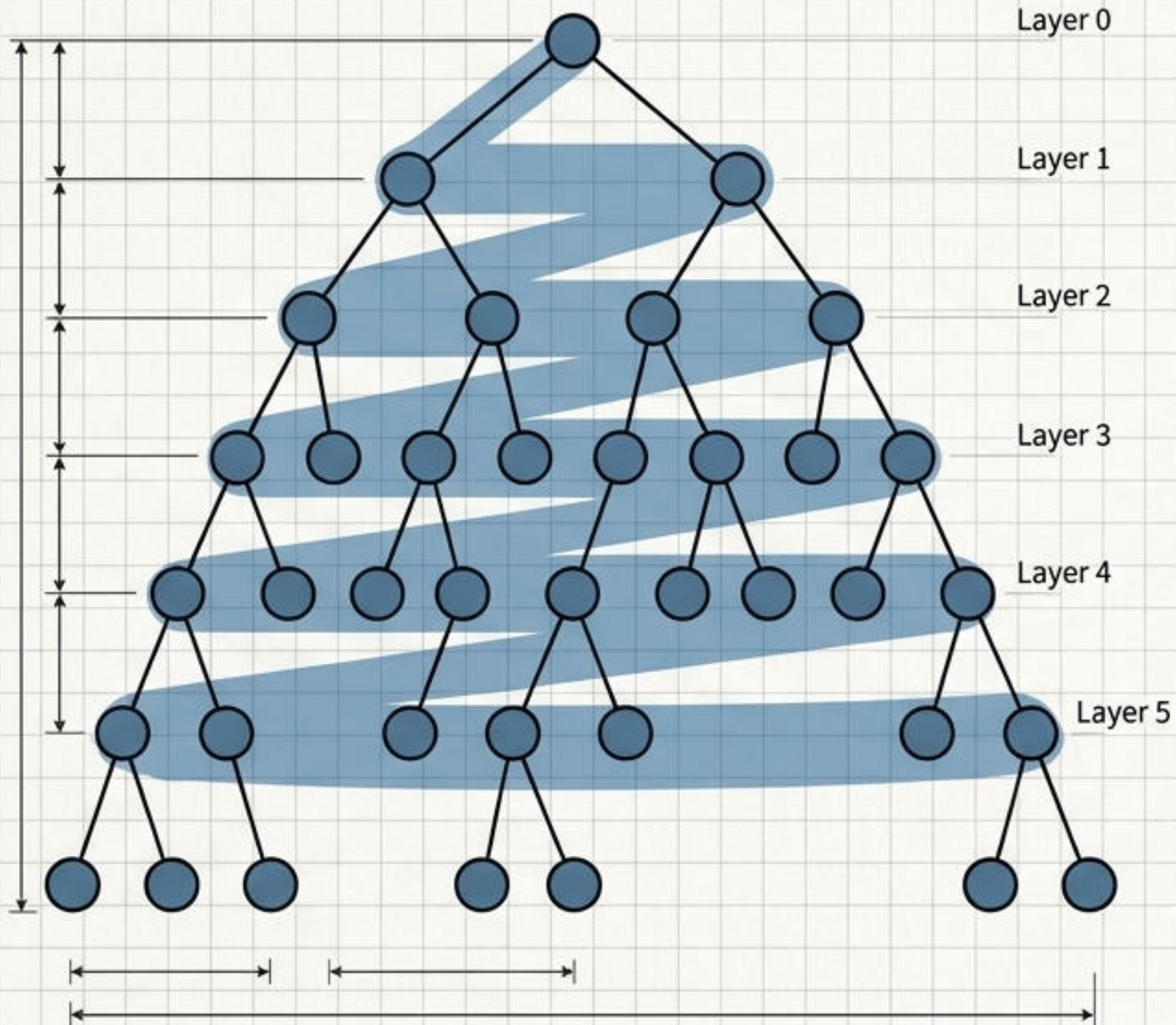


זיכרון (Space Complexity)

כמה צמתים
האלגוריתם חייב
לשמור בזיכרון
בו-זמנית?

הערה: מורכבות הזמן והזיכרון נמדדות במונחים של m , d , m .

חיפוש לרוחב (Breadth-First Search) – החוקר הזהיר



Diagnostic Data Card

שלמות: כן (אם סופי)

אופטימליות: כן (אם עלות כל צעד שווה ל-1)

 $O(b^{d+1})$: IMT

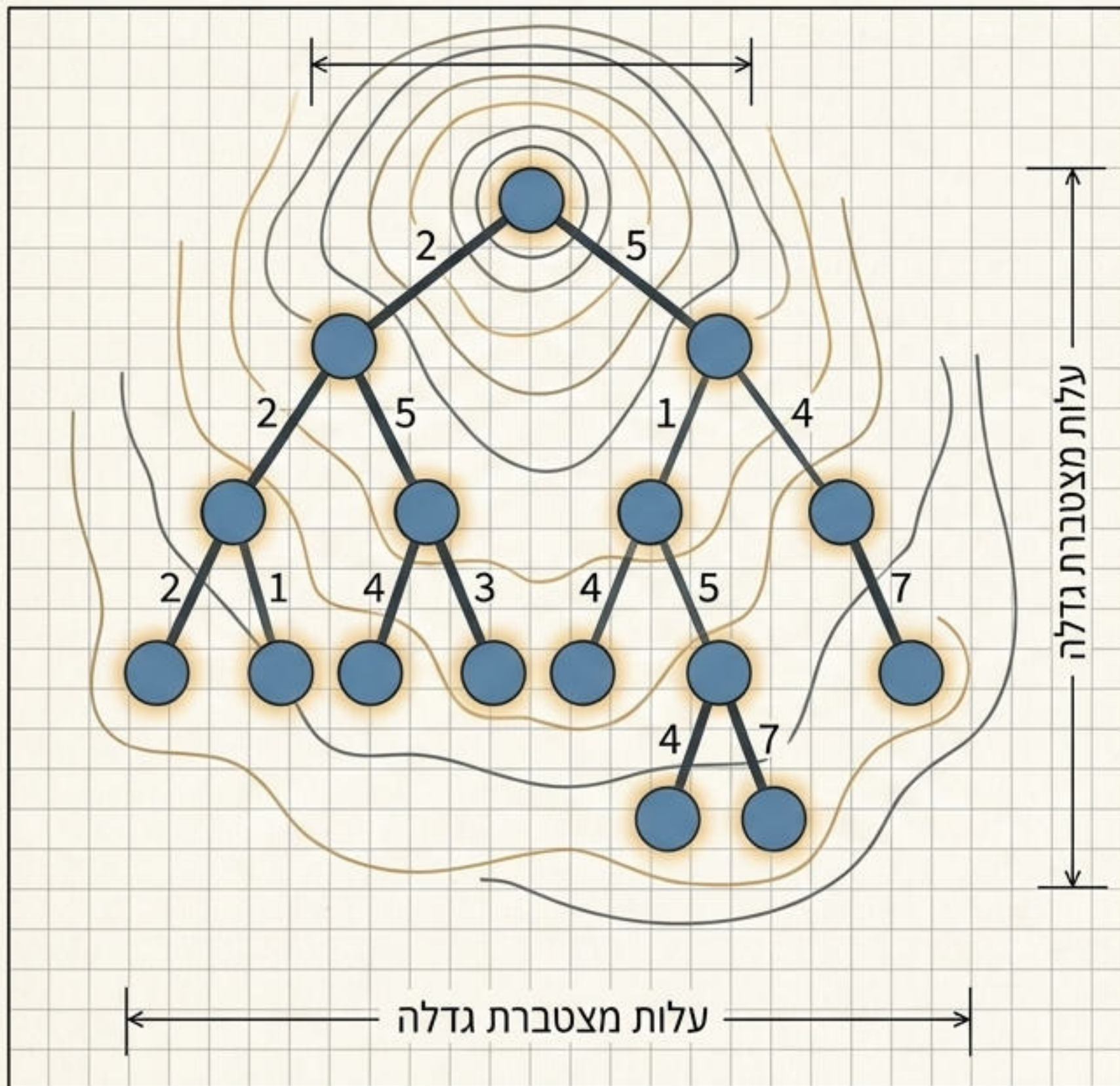
זיכרון: $O(b^{d+1})$

Callout Box →

הפגם הקטלני (זיכרון):

BFS שומר את כל העץ בזיכרון. המרחב גדל אקספוננציאלית. ייצור צמתים מהיר ימלא אלפי ג'יגה-בייט של זיכרון בתוך שעות ספורות, מה שהשיטה ללא ישימה לבעיות עמוקות.

חיפוש עלות אחידה (Uniform-Cost Search)



במקום להרחיב את הצומת הרדוד ביותר, UCS מרחיב את הצומת בעל העלות הכוללת הנמוכה ביותר עד כה.

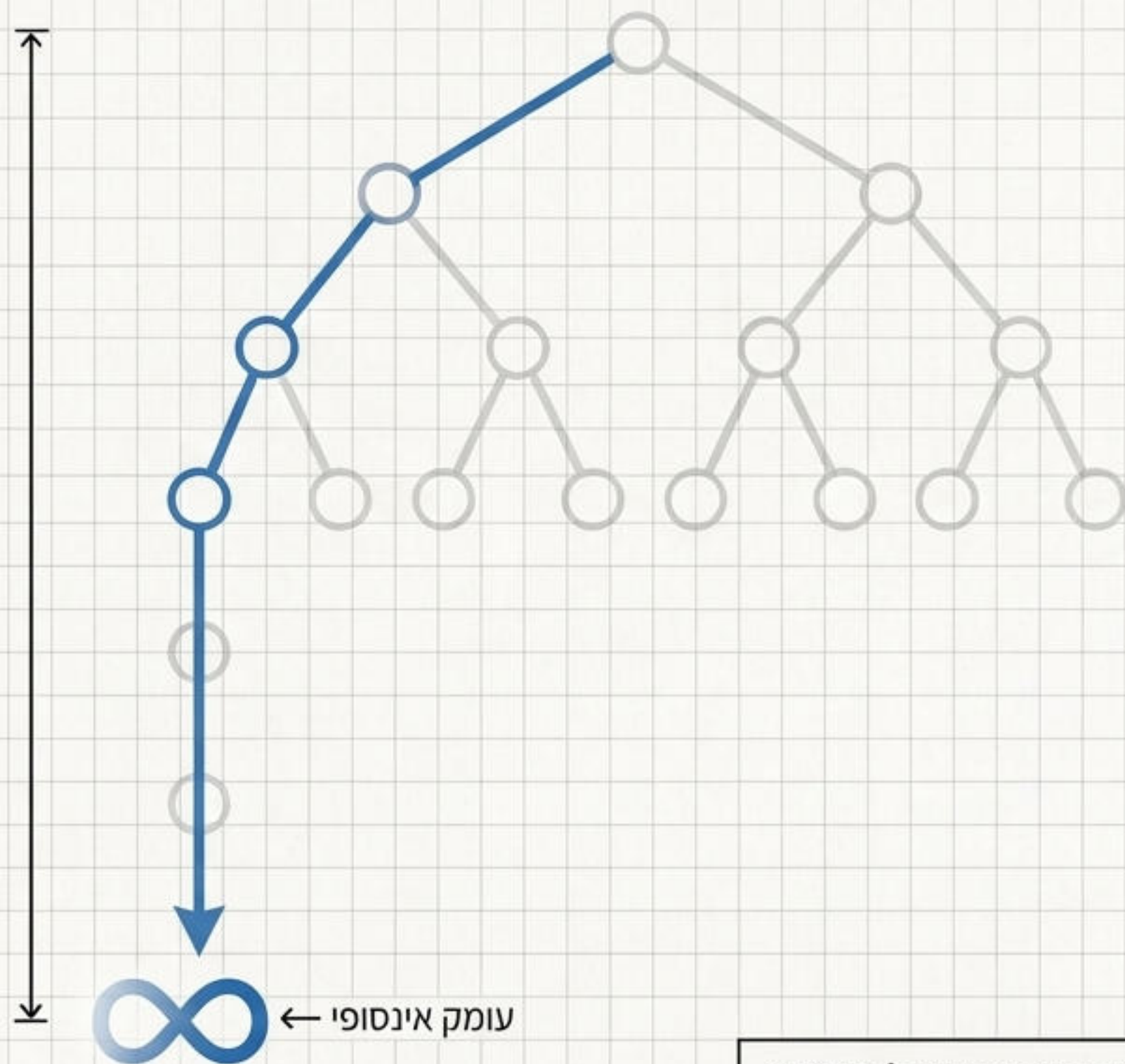
נתוני אבחון

✓	שלמות:	כן (אם עלות כל צעד גדולה מאפס)
✓	אופטימליות:	כן (מרחיב צמתים לפי סדר עלות עולה)
!	זמן וזיכרון:	$O(b^{[C^*/\epsilon]})$

שורה תחתונה

זהו שדרוג של BFS שלוקח בחשבון מחירים שונים, אך עדיין סובל מבעיית **פיצוץ הזיכרון האקספוננציאלי**.

חיפוש לעומק (Depth-First Search) – הצוללן הפזיז



Diagnostic Data Card	
שלמות: לא! (עלול להתקע בלולאות אינסופיות)	✗
אופטימליות: לא	
זמן: $O(b^m)$ (נורא אם m גדול בהרבה מ- d)	
זיכרון: $O(bm)$	✓

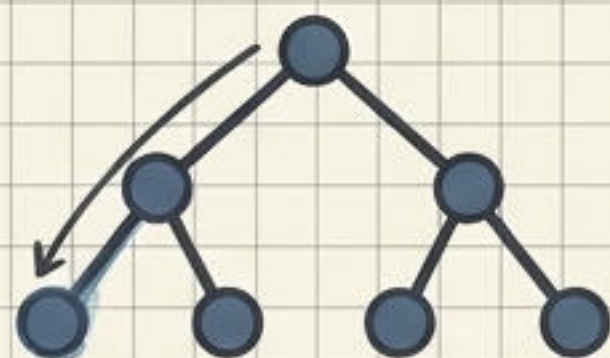
} יתרון עצום

נקודת האור (זיכרון ליניארי):

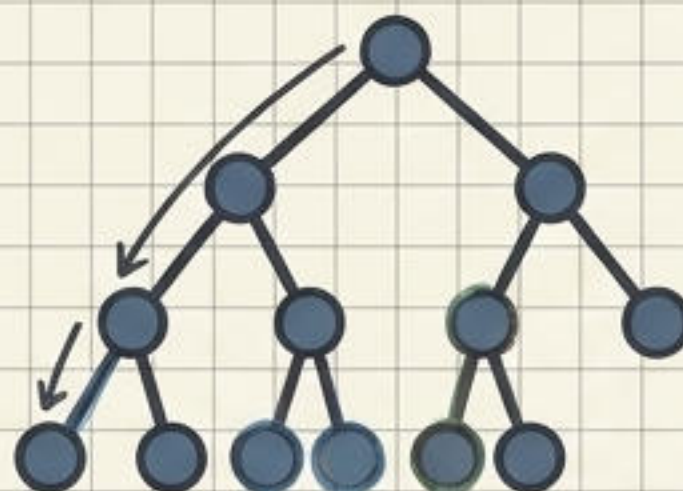
בניגוד ל-BFS, DFS שומר רק את המסלול הנוכחי. הוא דורש שבריר מזערי מהזיכרון.

האתגר: איך משלבים את צריכת הזיכרון המעולה של DFS עם הבטיחות של BFS?

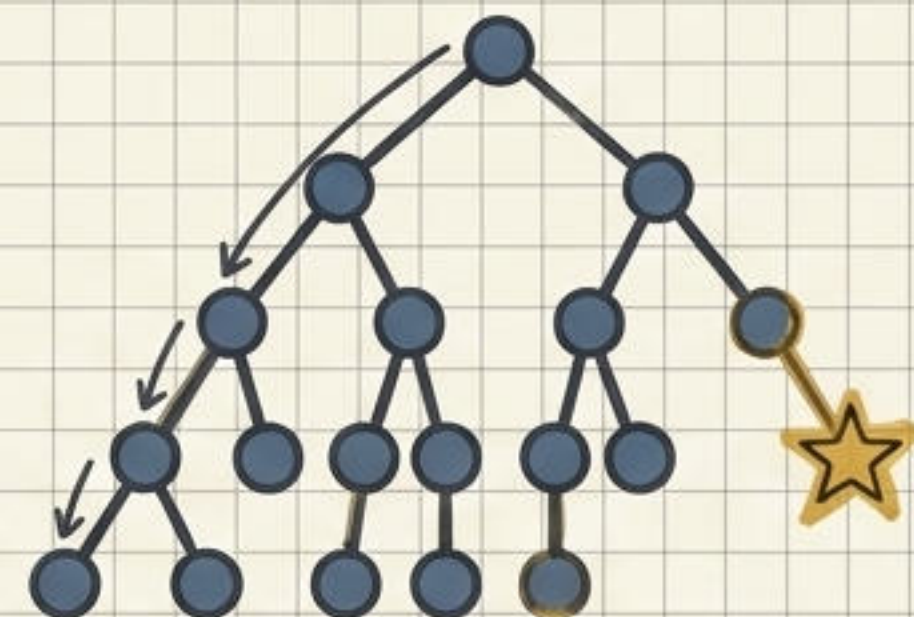
העמקה איטרטיבית (Iterative Deepening) IDS – המאסטר המשולב



שלב 1: חיפוש עד עומק 1



שלב 2: איפוס, וחיפוש עד עומק 2



שלב 3: איפוס, מציאת יעד בעומק 3

Diagnostic Data Card

זיכרון: $O(bd)$ ✓

שלמות: כן | אופטימליות: כן | זמן: $O(b^d)$

הטוב משני העולמות!

הפרדוקס: האם להתחיל מחדש זה בזבוז זמן?

לא! בעץ אקספוננציאלי, רוב מוחלט של העבודה נמצא ברמה התחתונה.

השוואה מספרית עבור $b=10, d=5$:

צמתים $N(IDS) = 123,450$

צמתים $N(BFS) = 1,111,100$

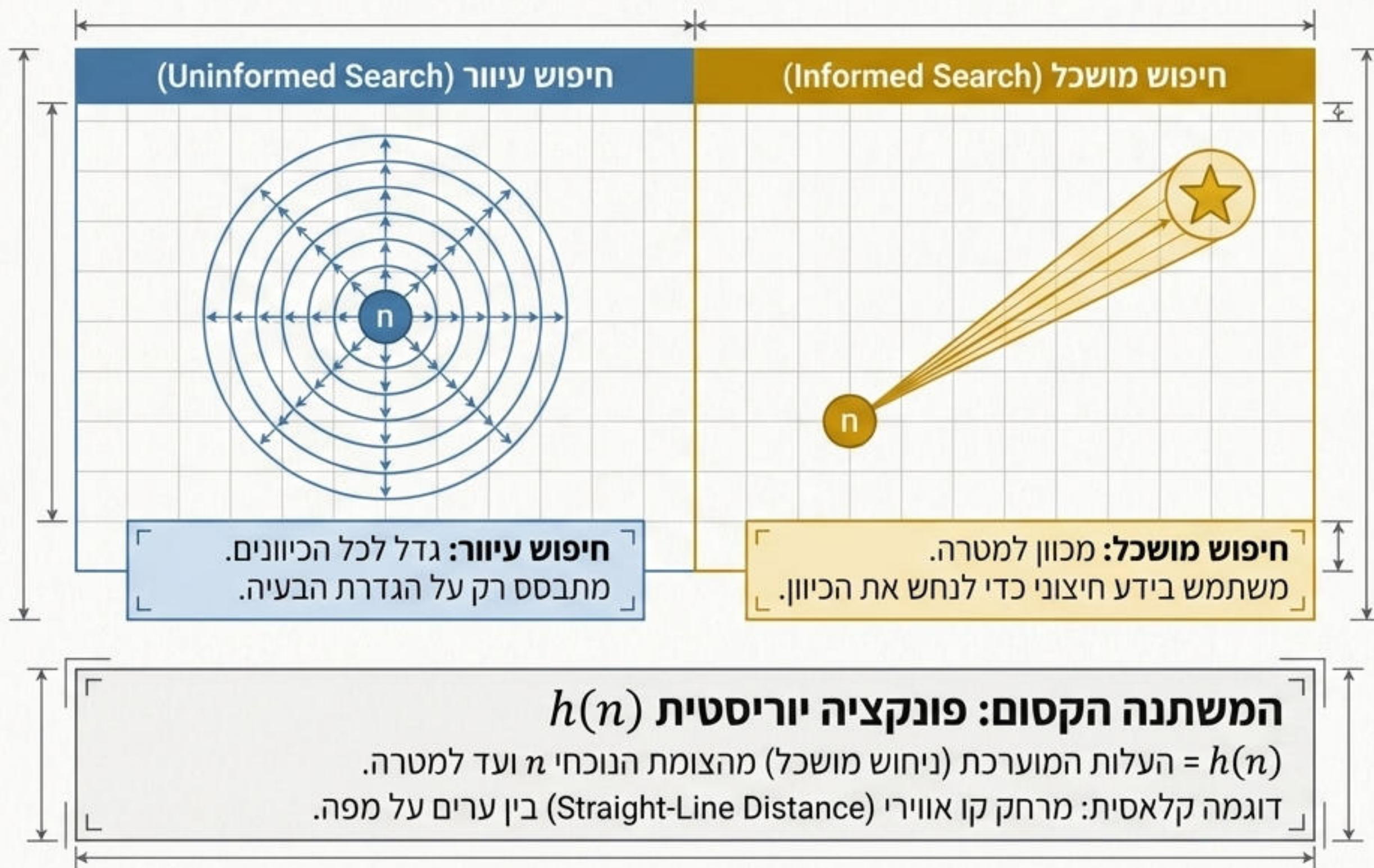
IDS בפועל יעיל יותר מ-BFS בזמן, ודורש זיכרון אפסי!

סיכום אסטרטגיות חיפוש עיוור (Uninformed Search)

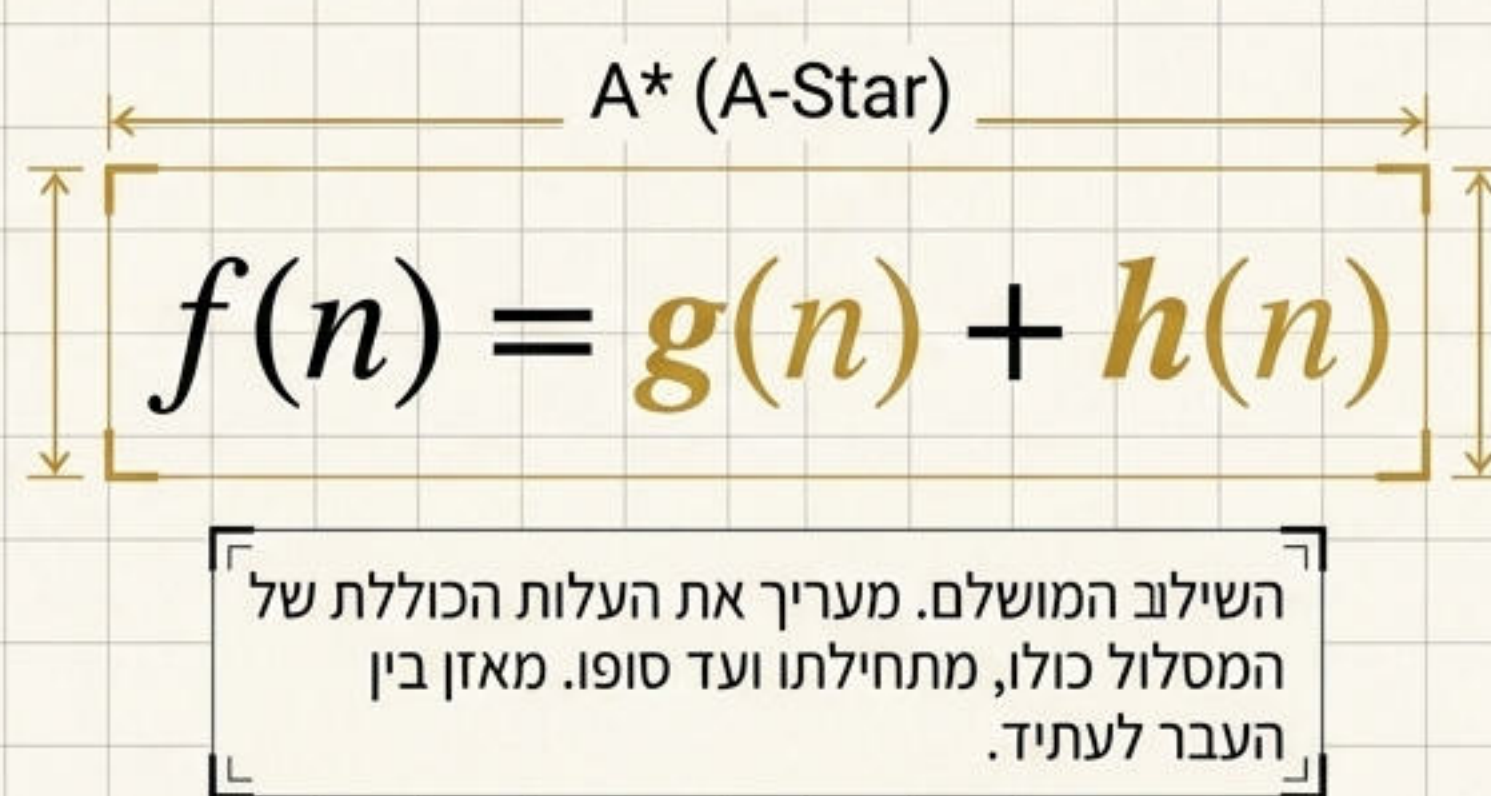
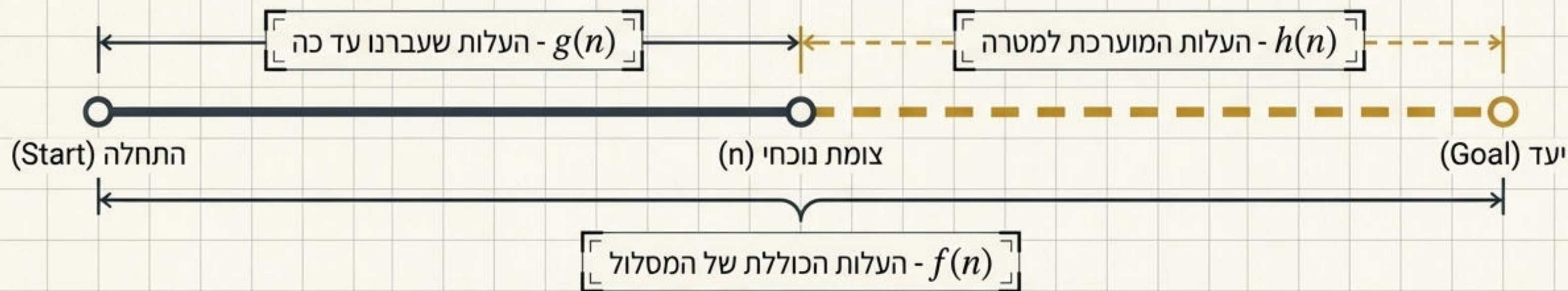
קריטריון	BFS	Uniform-Cost	DFS	Depth-Limited	Iterative Deepening (IDS)
שלמות?	כן	כן	לא	כן, אם $l \geq d$	כן
זמן	$O(b^{d+1})$	$O(b^{C^*/\epsilon})$	$O(b^m)$	$O(b^l)$	$O(b^d)$
זיכרון	$O(b^{d+1})$	$O(b^{C^*/\epsilon})$	$O(bm)$	$O(bl)$	$O(bd)$
אופטימליות?	כן	כן	לא	לא	כן

מסקנה: IDS היא שיטת החיפוש העיוור המומלצת ביותר. היא משלבת את היתרונות של BFS ו-DFS.

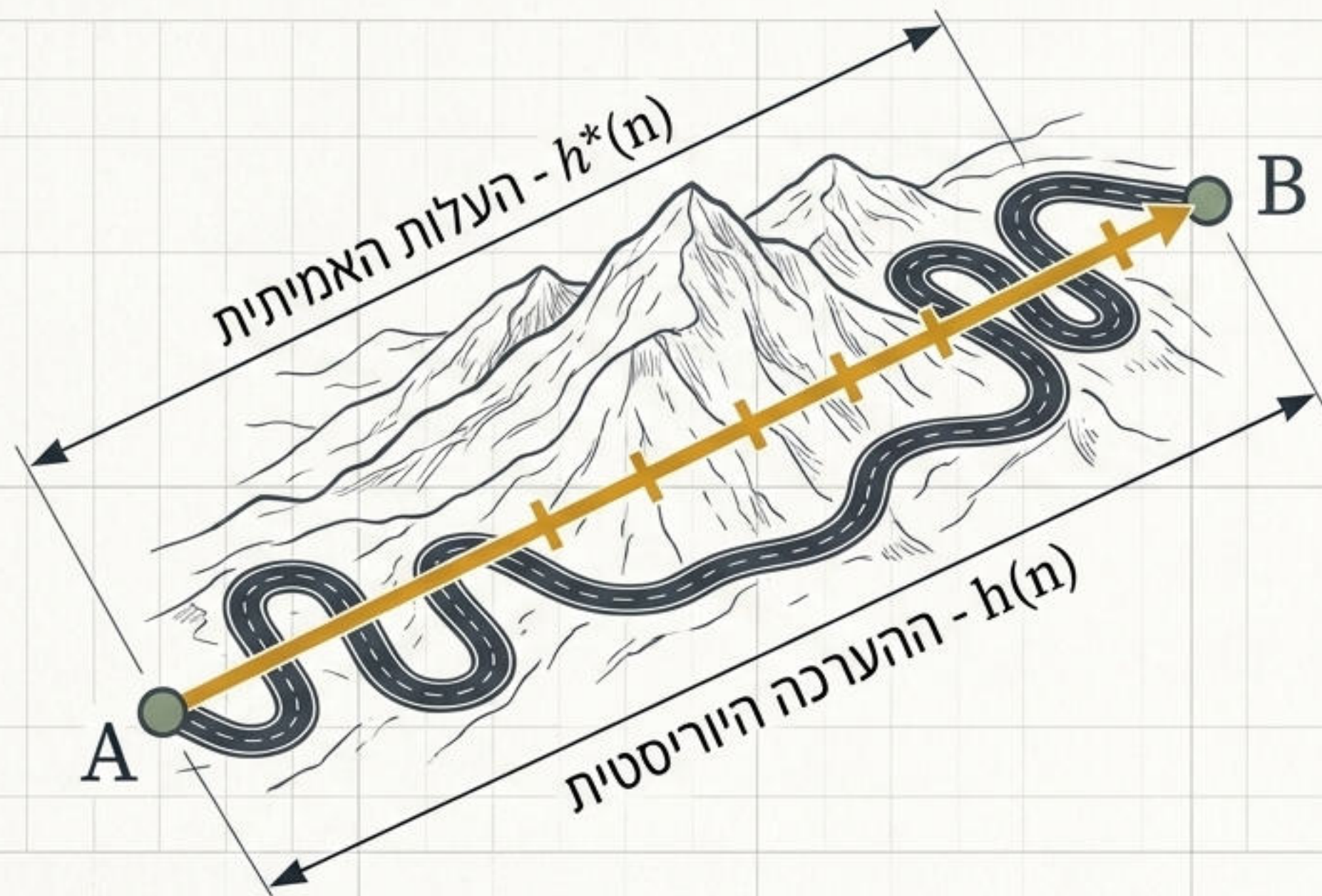
קפיצת המדרגה: מחיפוש עיוור לחיפוש מושכל (Informed Search)



בניית הפונקציה המנצחת: אלגוריתם A*



הסוד של A^* : יוריסטיקה קבילה (Admissible Heuristic)



חוק הברזל (The Golden Rule)

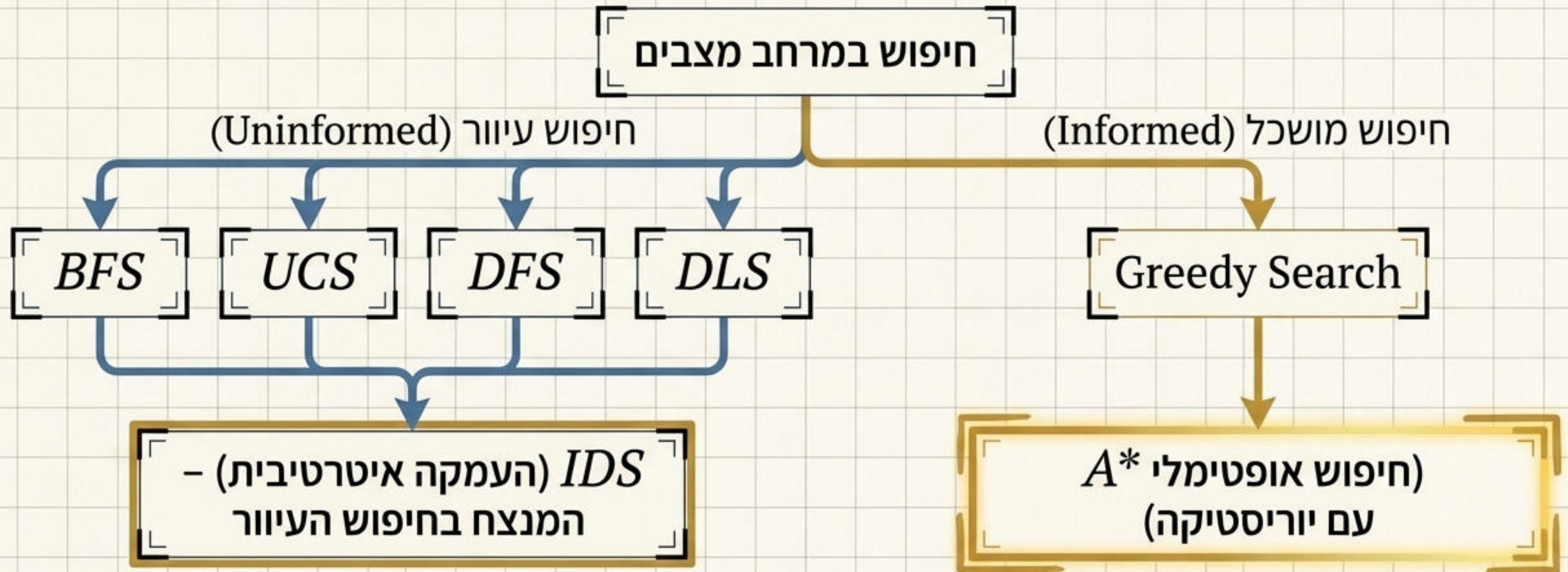
$$h(n) \leq h^*(n)$$

כדי ש- A^* יבטיח למצוא את המסלול האופטימלי, היוריסטיקה חייבת להיות קבילה: לעולם אין להפריז בהערכת המרחק. היא חייבת להיות תמיד אופטימית.

תכונות אלגוריתם A^* :

שלמות: כן | אופטימליות: כן | זיכרון: שומר את כל הצמתים בזיכרון (חסרון המשמש כפתח לאתגר הבא).

מפת הדרכים של אלגוריתמי החיפוש



מזל טוב! כעת יש לכם את הכלים להבין כיצד בינה מלאכותית מנווטת במרחבים מורכבים כדי למצוא את הפתרון הטוב ביותר.