# פתרון מטלה 12 – 2015א

### **שאלה** 1::

א.

**•** State space: A state is a tuple of the agent’s position, speed, and number of laps completed

• Start state: (1, 0, 0)

• Goal test: (1, 0, k)

ב.

No. For example, with six spaces left and velocity 3 or 4, the agent can park in four time steps (cost=4)

ג.

. Number of spaces left to go, divided by V , the max speed. In each time step, at most V spaces can be covered because V is an upper bound on the speed

ד.

No. The state space is infinite because the number of laps is in principle unbounded. Yes is also acceptable if you justify it by claiming the state space is finite because the number of laps should never exceed k.

ה.

Yes. All actions have uniform cost.

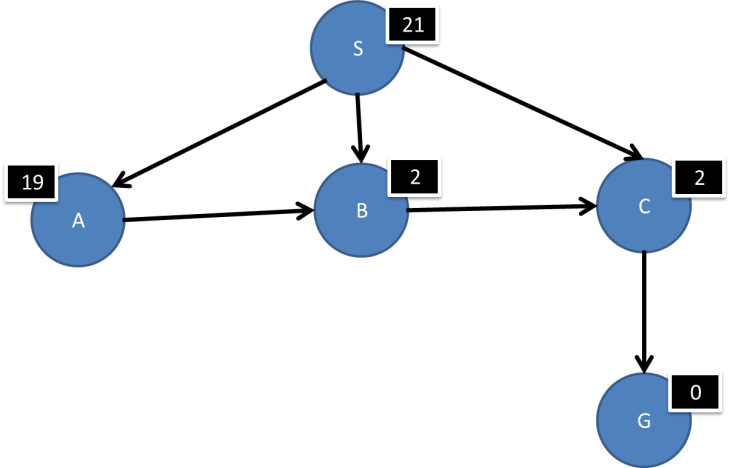
### **שאלה** 2:

הערכים על הצמתים מייצגים את ערך הפונקציה היוריסטית. לעומתם הערכים על גבי המעברים משקפים את המרחק\עלות.

הפונקציה היוריסטית קבילה כיוון שתמיד מייצגת מרחק קטן מהמרחק האמתי למטרה G.

הפונקציה אינה עקבית – למשל: המרחק האמתי בין A ל-B הוא 1 בעוד ההפרש לפי ערכי היוריסטיקה הוא 17.

הפתרון האופטימלי היה אמור להיות S -> A -> B -> C -> G. עם זאת, חיפוש A\* במקרה זה יפתח את הצמתים האחרים קודם כיוון שהערכים שלהם נמוכים מהמרחק מ-A.

****

### **שאלה 3 :**

1. כן. עבור כל צומת המחיר של המסלול האופטימלי אל צומת היעד G גדול או שווה לערך של h.
2. **DFS -** S → A → D → F → G
3. **BFS -** S → A → B → C → D → G
4. **IDS -** S → S → A → B → C → S → A → D → G
5. **Greedy Best First Search -** S→ C→ F->G
6. **Uniform Cost Search -** S,C,B,D,F,A,G.
7. **A\* Search-** S → C → F → G

שאלה 4 :

סעיף א.

הערכת המצב = מספר הקשתות בגרף שקצה אחד שלהן נמצא ב- והקצה השני שלהן ב-. נרצה להביא למינימום את הפונקציה היוריסטית לעיל.

סעיף ב.

יש דרכים שונות לבצע אלגוריתם טיפוס גבעה על פונקציה זו. לדוגמא, אפשר בשלב הראשון לחלק באופן רנדומלי את הקודקודים לשתי קבוצות שוות (או כמעט שוות) ואז כל שלב באלגוריתם יבחר להעביר קודקוד אחד מקבוצה אחת לשנייה, ואח"כ בחזרה - מהקבוצה השנייה לראשונה. בכל שלב באלגוריתם נבחר להעביר קודקוד *v* כלשהו לקבוצה השנייה שעבורו מתקיים: סכום הקשתות בינו לבין הקבוצה השנייה גדול מסכום הקשתות בינו לבין הקודקודים בקבוצה בה הוא נמצא. נבחר כל פעם את הקודקוד מהקבוצה שההפרש הזה הוא הגדול ביותר עבורו.

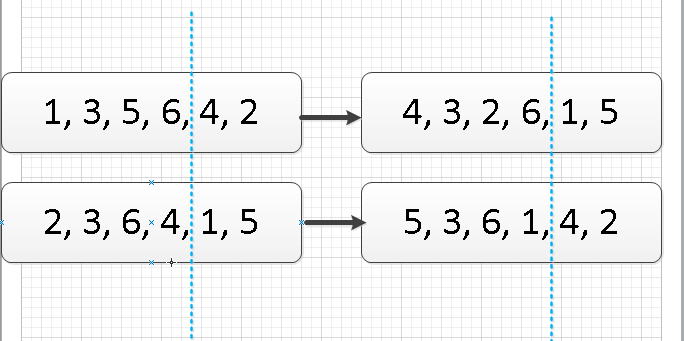
ההפרש הזה הוא ה"רווח" שהאלגוריתם יזכה בו אם נעביר את אותו קודקוד לקבוצה הנגדית. כלומר, אלגוריתם טיפוס גבעה על פונקציה זו יבחר בכל שלב של האלגוריתם את הקודקוד שהעברתו לצד השני תביא לירידה הגדולה ביותר במספר הקשתות העוברות בין שתי הקבוצות. אם לא קיים קודקוד כזה, משמע הגענו למינימום לוקאלי (אולי גם הגלובלי...) והאלגוריתם יעצור (אחרי שמספר הקודקודים יהיה מאוזן כמה שאפשר). באלגוריתם שתיארנו, מצב מתואר ע"י שתי רשימות של קודקודים השייכים לכל קבוצה וקבוצת המצבים השכנים תהיה התוצאה של העברת קודקוד v מהקבוצה עליה פועלים באותו שלב באלגוריתם, לקבוצה השנייה. כלומר, יהיו לנו כ-n/2 מצבים שכנים.

סעיף ג.

*כל פרט (מצב) צריך לתאר את החלוקה של הקודקודים בין שתי הקבוצות. אפשר לדוגמא למספר את הקודקודים השונים באופן הבא ואת הפרטים אפשר לקודד כרשימה סדורה של הקדקודים כך ש-n/2 הקודקודים בצד שמאל שייכים לקבוצה אחת, והשאר לשנייה. לדוגמא:*

*פונקצית ההתאמה, אשר נותנת ערכים גבוהים יותר למצבים טובים יותר, יכולה להיות ההפרש בין מספר הקשתות ה"פנימיות" (בין קודקודי כל קבוצה לבין עצמם) לבין מספר הקשתות החוצות. כלומר, נשאף להגדיל את מספר הקשתות הפנימיות ולהקטין את מספר הקשתות החוצות.*

*להלן דוגמא לפעולת ההצלבה:*

**

*שלא כמו בדוגמא המובאת בספר, על מנת שפעולת ההצלבה תהיה חוקית, יש מלבד ההצלבה לבצע תהליך של החלפה בכל פרט כדי שיהיה חוקי (כל קודקוד יופיע בו פעם אחת, חזרות אסורות).*

*פעולת המוטציה תהיה גם היא מעט שונה מזו המוצגת בדוגמא בספר, שכן לא נוכל לשנות ערך של איבר בודד ונצטרך להחליף 2 איברים זה בזה (בדומה לתהליך שעשינו בהצלבה לעיל). לכן נבחר אקראית איבר מהחצי השמאלי של הקידוד ואיבר מחציו הימני ונחליף ביניהם.*

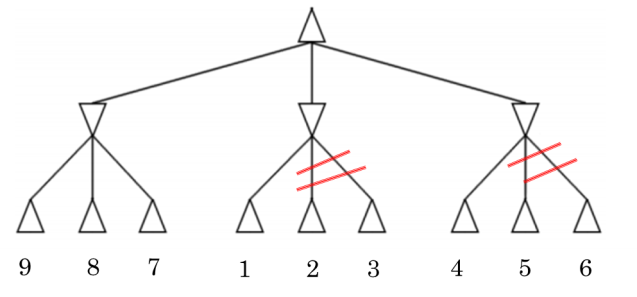
*סעיף ד*

*הדמיית חישול לבעיה זו יכולה להתבסס על האלגוריתם של טיפוס גבעה שהוצג בסעיף ב' עם השינוי הבא: האלגוריתם יגריל בכל שלב קדקוד. אם ההעברה של אותו קודקוד לקבוצה השנייה תקטין את מספר הקשתות החוצות, אזי הקודקוד יועבר. אם לא, אז הקודקוד עשוי עדיין לעבור לקבוצה השנייה בסבירות שתקטן עם התקדמות האלגוריתם.*

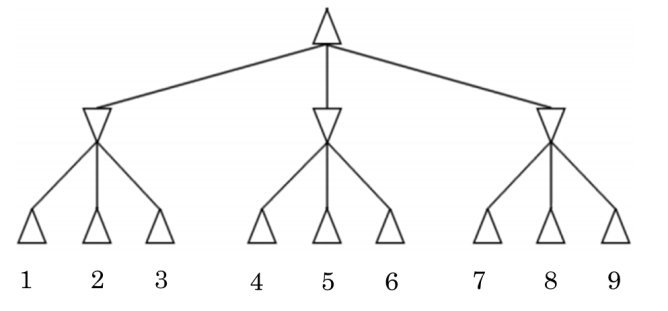
*אלגוריתם הדמיית חישול ואלגוריתמים גנטיים עדיפים על פני אלגוריתם טיפוס גבעה, שעלול להיתקע במינימום לוקאלי. אלגוריתמים אלו, לא תלויים בבחירה הרנדומאלית הראשונית כמו "טיפוס בגבעה", והם מאפשרים שינוי ושיפור ע"י שילוב של אקראיות ובחירה מושכלת בטוב. מבין שניהם, נראה לי שהאלגוריתם הגנטי עדיף, כי הוא מתחיל ממספר מצבים ראשונים (מוגרלים) שעשויים להיות מאד שונים האחד מהשני ומתפתח עליהם במקביל בעוד שהאלגוריתם של הדמיית חישול מפתח מצב אחד.*

שאלה 5 :

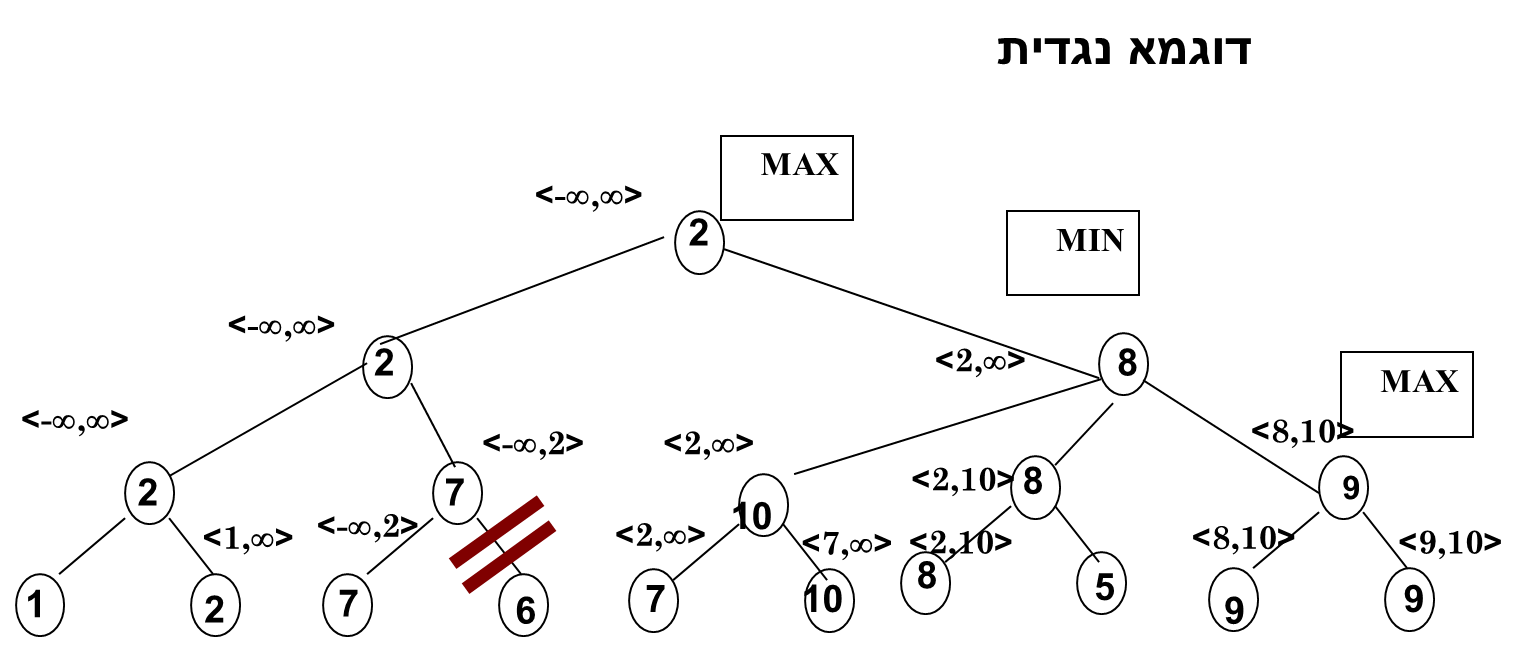
א.



ב.



ג. דוגמה נגדית:



שאלה 6 :

1. יש 5 משתנים וכל אחד מהם יכול לקבל 8 ערכים לכן 85.
2. d,i,j,e.

בתשובות לסעיפים הבאים רשום A במקום D ו-W במקום Y.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | A8 | A9 | A10 | A11 | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| H | A8 | A9 | A10 | ~~A11~~ | W8 | W9 | W10 | ~~W11~~ |
| P | A8 | A9 | A10 | A11 | W8 | W9 | W10 | W11 |
| S | A8 | A9 | ~~A10~~ | ~~A11~~ | W8 | W9 | ~~W10~~ | ~~W11~~ |
| T | A8 | A9 | A10 | ~~A11~~ | W8 | W9 | W10 | ~~W11~~ |

1. לאחר מחיקת הערכים בסעיף ג' נבחר S=D9. יש למחוק את כל הערכים שיימחקו באמצעות בדיקה קדימה:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | A8 | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| H | ~~A8~~ | ~~A9~~ | A10 | ~~A11~~ | ~~W8~~ | ~~W9~~ | W10 | ~~W11~~ |
| P | A8 | ~~A9~~ | A10 | A11 | W8 | ~~W9~~ | W10 | W11 |
| S | ~~A8~~ | A9 | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| T | ~~A8~~ | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | W10 | ~~W11~~ |

1. יוריסטיקת MRV תבחר את המשתנה שמספר הערכים האפשריים שנותרו לו מינימלי ולכן תבחר את F .

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | A8 |  | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| H | ~~A8~~ |  | ~~A9~~ | A10 | ~~A11~~ | ~~W8~~ | ~~W9~~ | W10 | ~~W11~~ |
| P | ~~A8~~ |  | ~~A9~~ | A10 | A11 | W8 | ~~W9~~ | W10 | W11 |
| S | ~~A8~~ |  | A9 | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| T | ~~A8~~ |  | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | W10 | ~~W11~~ |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | A8 | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| H | ~~A8~~ | ~~A9~~ | A10 | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| P | ~~A8~~ | ~~A9~~ | ~~A10~~ | ~~A11~~ | W8 | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| S | ~~A8~~ | A9 | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | ~~W10~~ | ~~W11~~ |
| T | ~~A8~~ | ~~A9~~ | ~~A10~~ | ~~A11~~ | ~~W8~~ | ~~W9~~ | W10 | ~~W11~~ |

1. עקביות קשת גרמה למחיקת יותר ערכים שכן היא בודקת את כל הקשתות (בודקת עקביות של כל זוגות המשתנים) .

Arc consistency removes more values. It’s because AC checks consistency between any pair of variables, while FC only checks the relationship between pairs of assigned and unassigned variables.

1. F: A8 H: A10 P: W8 S: A9 T: W10

# פתרון מטלה 17

### **שאלה** 1:

סעיף א.

סעיף ב

**שלב 0** : איפוס כל המצבים.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***שלב*** | ***חדר רחצה*** | ***מטבח*** | ***חדר שינה*** | ***חדר אוכל*** | ***תחת מתקפה*** | ***מת*** |
| *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| *1* | *4* | *10* | *0* | *2* | -50 | *0* |
| *2* | *7* | *11.6* | *1.6* | *5* | -50 | 0 |
| *3* | 7.8 | 13.1 | 3.1 | 5.8 | -50 | 0 |
| *4* | 8.55 | 13.5 | 3.5 | 6.55 | -50 | 0 |
| *5* | 8.75 | 13.875 | 3.875 | 6.75 | -50 | 0 |
| *6* | 8.9375 | 13.975 | 3.975 | 6.9375 | -50 | 0 |

**שלב 1** : בשלב הראשון יש להכניס את התגמולים של כל אחד מהמצבים לטבלה.

**שלב 2** :   
חיפוש התוחלת המקסימלית עבור חדר הרחצה :

H : 0.6 \* 10 + 0.4 \* 0 = 6 = Max

V : 0.4 \* 10 + 0.6 \* 0 = 4

S : 0.75 \* 4 + 0.25 \* (-50) = -9.5

ערך התועלת עבור חדר הרחצה :

U = 4 + 0.5 \* 6 = 7

חיפוש התוחלת המקסימלית עבור המטבח :

H : 0.6 \* 4 + 0.4 \* 2 = 3.2 = Max

V : 0.4 \* 4 + 0.6 \* 2 = 1.92

S : 0.75 \* 10 + 0.25 \* (-50) = -5

חישוב ערך התועלת עבור המטבח :

U = 10 + 0.5 \* 3.2 = 11.6

חיפוש התוחלת המקסימלית עבור חדר השינה :

H : 0.4 \* 4 + 0.6 \* 2 = 2.8

V : 0.6 \* 4 + 0.4 \* 2 = 3.2 = Max

S : 0.75 \* 0 + 0.25 \* (-50) = -12.5

חישוב ערך התועלת עבור חדר השינה :

U = 0 + 0.5 \* 3.2 = 1.6

חיפוש התוחלת המקסימלית עבור חדר האוכל :

H : 0.6 \* 10 + 0.4 \* 0 = 6 = Max

V : 0.4 \* 10 + 0.6 \* 0 = 4

S : 0.75 \* 2 + 0.25 \* (-50) = -11

חישוב ערך התועלת עבור חדר האוכל :

U = 2 + 0.5 \* 6 = 5

חישוב ערך התועלת עבור תחת מתקפה :

U = -50 + 0.5 \* (1 \* 0) = -50

חישוב ערך התועלת עבור מת :

U = 0 + 0.5 \* (1 \* 0) = 0

**שלב 3** :   
חיפוש התוחלת המקסימלית עבור חדר הרחצה :

H : 0.6 \* 11.6 + 0.4 \* 1.6 = 7.6 = Max

V : 0.4 \* 11.6 + 0.6 \* 1.6 = 5.6

S : 0.75 \* 7 + 0.25 \* (-50) = -7.25

חישוב ערך התועלת עבור חדר הרחצה :

U = 4 + 0.5 \* 7.6 = 7.8

חיפוש התוחלת המקסימלית עבור המטבח :

H : 0.6 \* 7+ 0.4 \* 5 = 6.2 = Max

V : 0.4 \* 7 + 0.6 \* 5 = 5.8

S : 0.75 \* 11.6 + 0.25 \* (-50) = -3.8

חישוב ערך התועלת עבור המטבח :

U = 10 + 0.5 \* 6.2 = 13.1

חיפוש התוחלת המקסימלית עבור חדר השינה :

H : 0.4 \* 7 + 0.6 \* 5 = 5.8

V : 0.6 \* 7 + 0.4 \* 5 = 6.2 = Max

S : 0.75 \* 1.6 + 0.25 \* (-50) = -11.3

חישוב ערך התועלת עבור חדר השינה :

U = 0 + 0.5 \* 6.2 = 3.1

חיפוש התוחלת המקסימלית עבור חדר האוכל :

H : 0.6 \* 11.6 + 0.4 \* 1.6 = 7.6 = Max

V : 0.4 \* 11.6 + 0.6 \* 1.6 = 5.6

S : 0.75 \* 5 + 0.25 \* (-50) = -8.75

*חישוב ערך התועלת עבור חדר האוכל:*

U = 2 + 0.5\*7.6 = 5.8

*חישוב כרך התועלת עבור תחת מתקפה:*

U = -50 + 0.5 \* (1 \* 0) = -50

*חישוב רך התועלת עבור מת:*

U = 0 + 0.5 \* (1 \* 0) = 0

**שלב 4** :   
חישוב ערך התועלת עבור חדר הרחצה :

U = 4 + 0.5 \* (0.6 \* 13.1 + 0.4 \* 3.1) = 8.55

חישוב ערך התועלת עבור המטבח :

U = 10 + 0.5 \* (0.6 \* 7.8 + 0.4 \* 5.8) = 13.5

חישוב ערך התועלת עבור חדר השינה :

U = 0 + 0.5 \* (0.6 \* 7.8 + 0.4 \* 5.8 ) = 3.5

חישוב ערך התועלת עבור חדר האוכל :

U = 2 + 0.5 \* (0.6 \* 13.1 + 0.4 \* 3.1) = 6.55

ערך התועלת עבור תחת מתקפה ומת לא משתנה

**שלב 5** :   
חישוב ערך התועלת עבור חדר הרחצה :

U = 4 + 0.5 \* (0.6 \* 13.5 + 0.4 \* 3.5) = 8.75

חישוב ערך התועלת עבור המטבח :

U = 10 + 0.5 \* (0.6 \* 8.55 + 0.4 \* 6.55) = 13.875

חישוב ערך התועלת עבור חדר השינה :

U = 0 + 0.5 \* (0.6 \* 8.55 + 0.4 \* 6.55 ) = 3.875

חישוב ערך התועלת עבור חדר האוכל :

U = 2 + 0.5 \* (0.6 \* 13.5 + 0.4 \* 3.5) = 6.75

ערך התועלת עבור תחת מתקפה ומת לא משתנה

**שלב 6** :   
חישוב ערך התועלת עבור חדר הרחצה :

U = 4 + 0.5 \* (0.6 \* 13.875 + 0.4 \* 3.875) = 8.9375

חישוב ערך התועלת עבור המטבח :

U = 10 + 0.5 \* (0.6 \* 8.75 + 0.4 \* 6.75) = 13.975

חישוב ערך התועלת עבור חדר השינה :

U = 0 + 0.5 \* (0.6 \* 8.75 + 0.4 \* 6.75 ) = 3.975

חישוב ערך התועלת עבור חדר האוכל :

U = 2 + 0.5 \* (0.6 \* 13.875 + 0.4 \* 3.875) = 6.9375

ערך התועלת עבור תחת מתקפה ומת לא משתנה

להלן המדיניות האופטימלית עבור כל אחד מהמצבים :  
חדר רחצה – לנוע אופקית (H)

מטבח – לנוע אופקית(H)

חדר שינה – לנוע אנכית (V)

חדר אוכל – לנוע אופקית (H)

תחת מתקפה – למות

מת – להישאר מת

שאלה **2**:

נשתמש באלגוריתם DTL ע"מ לבנות את העץ .

S = Class

P(Class=p) = 4/7

P(Class=n) = 3/7

Entropy(S) = Entropy(Class) = -4/7\*log(4/7)-3/7\*log(3/7) = 0.985228

כעת נבחן את התכונות F1,F2,F3,F4,F5 ובדוק עבור איזה תכונה נקבל את ה Gain הכי גבוה:

F1=true : Entropy(F1\_true) = -3/4\*log(3/4)-1/4\*log(1/4) = 0.811278

F1=false : Entropy(F1\_false) = -1/3\*log(1/3)-2/3\*log(2/3) = 0.9183

Gain(F1) = Entropy(S) – 4/7\*Entropy(F1\_true) – 3/7\*Entropy(F1\_false) = 0.985228-4/7\*0.811278-3/7\*0.9183 = 0.12808

F2=true : Entropy(F2\_true) = -1/4\*log(1/4)-3/4\*log(3/4) = 0.81128

F2=false : Entropy(F2\_false) = -3/3\*log(3/3) = 0

Gain(F2) = Entropy(S) – 4/7\*Entropy(F2\_true) – 3/7\*Entropy(F2\_false) = 0.985228-4/7\*0.81128-3/7\*0 = 0.52164

F3=true : Entropy(F3\_true) = -2/3\*log(2/3)-1/3\*log(1/3) = 0.9183

F3=false : Entropy(F3\_false) = -2/4\*log(2/4) -2/4\*log(2/4) = 1

Gain(F3) = Entropy(S) – 3/7\*Entropy(F3\_true) – 4/7\*Entropy(F3\_false) = 0.985228-3/7\*0.9183-4/7\*1 = 0.02024

F4=true : Entropy(F4\_true) = -2/4\*log(2/4) -2/4\*log(2/4) = 1

F4=false : Entropy(F4\_false) = -2/3\*log(2/3)-1/3\*log(1/3) = 0.9183

Gain(F4) = Entropy(S) – 4/7\*Entropy(F4\_true) – 3/7\*Entropy(F4\_false) = 0.985228-4/7\*1-3/7\*0.9183 = 0.02024

F5=true : Entropy(F5\_true) = -1/3\*log(1/3)-2/3\*log(2/3) = 0.9183

F5=false : Entropy(F5\_false) = -3/4\*log(3/4)-1/4\*log(1/4) = 0.811278

Gain(F5) = Entropy(S) – 3/7\*Entropy(F5\_true) – 4/7\*Entropy(F5\_false) = 0.985228-3/7\*0.9183-4/7\*0.811278 = 0.12808

קיבלנו כי הGain הגבוהה ביותר הוא 0.52164 Gain(F2) = ולכן זו תהיה התכונה של השורש -

true

false

דוגמא1 = p

דוגמא5 =n

דוגמא6 = n

דוגמא7 = n

p

דוגמא2 = p

דוגמא3 =p

דוגמא4 = p

כעת, נרצה להוסיף עוד צומת החלטה עבור כל הדוגמאות שעבורן F2=true (נוריד את התכונה F2 ) , נגדיר קבוצה זו

כS\_2 :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | F1 | F3 | F4 | F5 | Class |
| דוגמא1 | true | false | false | false | p |
| דוגמא5 | false | false | false | false | n |
| דוגמא6 | true | false | true | true | n |
| דוגמא7 | false | true | true | true | n |

נחשב את ה Entropy של כל הקבוצה -

Entropy(S\_2) = Entropy(Class) = -1/4\*log(1/4,2)-3/4\*log(3/4,2) = 0.811278

כעת נבחן את התכונות F1,F3,F4,F5 ובדוק עבור איזה תכונה נקבל את ה Gain הכי גבוה:

F1=true : Entropy(F1\_true) = -1/2\*log(1/2,2)-1/2\*log(1/2,2) = 1

F1=false : Entropy(F1\_false) =-0 -2/2\*log(2/2,2) = 0

Gain(F1) = Entropy(S\_2) – 2/4\*Entropy(F1\_true) – 2/4\*Entropy(F1\_false) = 0.811278-2/4\*0.30103-2/4\*0 = 0.31128

F3=true : Entropy(F3\_true) = -0-1/1\*log(1/1,2) = 0

F3=false : Entropy(F3\_false) = -1/3\*log(1/3,2) -2/3\*log(2/3,2) = 0.9183

Gain(F3) = Entropy(S\_2) – 1/4\*Entropy(F3\_true) – 3/4\*Entropy(F3\_false) = 0.811278-1/4\*0-3/4\*0.9183 = 0.12255

F4=true : Entropy(F4\_true) = -0 -2/2\*log(2/2,2) = 0

F4=false : Entropy(F4\_false) = -1/2\*log(1/2,2)-1/2\*log(1/2,2) = 1

Gain(F4) = Entropy(S\_2) – 2/4\*Entropy(F4\_true) – 2/4\*Entropy(F4\_false) = 0.811278-2/4\*0-2/4\*1 = 0.31128

F5=true : Entropy(F5\_true) = -0 -2/2\*log(2/2,2) = 0

F5=false : Entropy(F5\_false) = -1/2\*log(1/2,2)-1/2\*log(1/2,2) = 1

Gain(F5) = Entropy(S\_2) – 2/4\*Entropy(F5\_true) – 2/4\*Entropy(F5\_false) = 0.811278-2/4\*0-2/4\*1 = 0.31128

קיבלנו כי הGain הגבוהה ביותר הוא 0.31128 Gain(F1), Gain(F4) , Gain(F5) = ולכן נבחר ב F1 כצומת ההחלטה הבאה, כעת העץ יראה כך :

true

false

p

false

דוגמא5 =n

דוגמא7 = n

דוגמא1 = p

דוגמא6 = n

true

כעת, נרצה להוסיף עוד צומת החלטה עבור כל הדוגמאות שעבורן F1=true (נוריד את התכונה F1 ) , נגדיר קבוצה זו

כS\_3 :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | F3 | F4 | F5 | Class |
| דוגמא1 | false | false | false | p |
| דוגמא6 | false | true | true | n |

נחשב את ה Entropy של כל הקבוצה -

Entropy(S\_3) = Entropy(Class) = -1/2\*log(1/2,2)-1/2\*log(1/2,2) = 1

כעת נבחן את התכונות F3,F4,F5 ונבדוק עבור איזה תכונה נקבל את ה Gain הכי גבוה:

F3=true : Entropy(F3\_true) = 0

F3=false : Entropy(F3\_false) = -1/2\*log(1/2,2) -1/2\*log(1/2,2) = 1

Gain(F3) = Entropy(S\_3) – 0/2\*Entropy(F3\_true) – 2/2\*Entropy(F3\_false) = 1-0-2/2\*1 = 0

F4=true : Entropy(F4\_true) = 0 -1/1\*log(1/1,2) = 0

F4=false : Entropy(F4\_false) = -1/1\*log(1/1,2) - 0 = 0

Gain(F4) = Entropy(S\_3) – 1/2\*Entropy(F4\_true) – 1/2\*Entropy(F4\_false) = 1

F5=true : Entropy(F5\_true) = 0 -1/1\*log(1/1,2) = 0

F5=false : Entropy(F5\_false) = -1/1\*log(1/1,2) - 0 = 0

Gain(F5) = Entropy(S\_3) – 1/2\*Entropy(F5\_true) – 1/2\*Entropy(F5\_false) = 1

קיבלנו כי הGain הגבוהה ביותר הוא , Gain(F4) , Gain(F5) = 1 ולכן נבחר ב F4 כצומת ההחלטה הבאה, כעת העץ הסופי יראה כך :

false

true

p

true

false

n

true

false

n

p

ב. דוגמא8 תסווג כשייכת ל-class : P.