

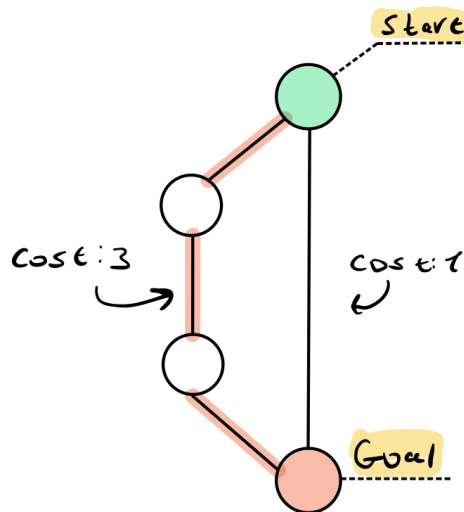
דוח מטלה 1 – בינה מלכותית

1.1 פאקמן לא ביקר בכל הצמתים שנחקרו כמצופה מכוון שקיימים צמתים ללא מוצא וכוון שפאקמן לא חוזר על עקבותיו אינו יבקר בכל צומת ש DFS מצא.

1.2 זהו איננו הפתרון בעל העלות הנמוכה ביותר שכן קיים מסלול קצר יותר ליעד. DFS למעשה חוקר לעומק תחילה את בנים (צמתים) השמאליים ורק לאחר מכן חוזר אחורה ברקורסיה לבדוק את השאר משמאל לימין ולכן גם אם יש בן שמגיע ישירות ליעד אם יש מסלול שמתחיל מהבן השמאלי הוא יבחר בו.

באדום המסלול ש DFS יחזיר:

- דוגמא:



4.1 להלן תוצאות אלגוריתמי החיפוש ב – openMaze

DFS: expanded: 806, cost: 298

אכן צפוי ש DFS יפתח הרבה מאד צמתים שכן הוא פשוט הולך באופן 'עיוור' לצומת העמוקה ביותר וממשיך ברקורסיה.

BFS: expanded: 682, cost: 54

BFS כמובן בסוף מוצא את כל המסלולים הקצרים ביותר ממצב ההתחלה. לכן פשוט נשלוף את המסלול הקצר ביותר של מצב היעד שמצאנו.

UCS: expanded: 682, cost: 54

UCS מחזיר אותו ערך כמו של BFS שכן עלות כל מעבר ממצב למצב היא 1 כברירת מחדל, לכן למעשה המסלול הזול ביותר הוא פשוט המסלול הקצר ביותר ולכן קיבלנו אותה תשובה כמו BFS

A-Star: expanded: 535, cost 54

AStar במקרה הזה מצא את המסלול הטוב ביותר שכן מקבל מידע קודם על מרחק מצב ממצב-היעד מה שמאפשר לו לא לחקור מצבים שמרוחקים ממצב היעד ולכן לעומת UCS לא נחקר את כל המצבים בעומק מסוים (כאשר מחיר מעבר מצומת לצומת הוא קבוע)

5.1 בחרתי לייצג מצב כ – tuple אשר מחזיק את המיקום כערך ראשון ואת הפינות

שעוד לא ביקרנו בהן כערך שני. כך שכאשר אנו במצב שמספר הפינות שלא ביקרנו בו שווה ל- 0 אנו יודעים כי זהו state goal.

- 6.1 הפונקציה ההיוריסטית שבחרתי לממש היא לחשב את המרחק-מנהטן מן הפינה הרחוקה ביותר שלא נחקרה. הפונקציה תחשב את המרחק בצעדים מן הפינה הרחוקה ביותר ותחזיר את תשובתה.
- נוכונות: נכונות הפונקציה נובעת מן התכונה הבאה: ככל שפאקמן מתקרב לפינה רחוקה יותר כך חוזר ערך נמוך יותר וכוון שבכל מקרה פאקמן יאלץ לעבור את המרחק הזה למעשה קיבלנו פונקציה שמחזירה כמה אנחנו קרובים לסיום.

להלן תוצאות הריצה:

```
Apple / ~ / De / Homework / AI / ex1 / master !2 > python pacman.py -l mediumCorners -p AStarCornersAgent -z 0.5
Path found with total cost of 106 in 2.5 seconds
Search nodes expanded: 857
Pacman emerges victorious! Score: 434
Average Score: 434.0
Scores: 434.0
Win Rate: 1/1 (1.00)
Record: Win
```

- 7.1 הפקונציה ההיוריסטית שבחרתי זהה לזו של 6.1 רק שהפעם אנו מחזירים את המרחק בצעדים של האוכל הרחוק ביותר
- נוכונות: זהה לזו של 6.1

```
Apple / ~ / De / Homework / A / ex1 / master !2 > python pacman.py -l trickySearch -p AStarFoodSearchAgent
Path found with total cost of 60 in 5.4 seconds
Search nodes expanded: 6021
Pacman emerges victorious! Score: 570
Average Score: 570.0
Scores: 570.0
Win Rate: 1/1 (1.00)
Record: Win
```

אציין ואוסיף כי בחרתי להוסיף את שורת הפקודה מכוון שעיצבתי את הטרמינל יפה ורציתי שתראו 😊