

Data oddania: \_\_\_\_\_

Ocena: \_\_\_\_\_

Maciej Pracucik 216869

Adam Józwiak 2167862

## Zadanie 1: Piętnastka

### 1. Cel

Celem zadania było napisanie programu rozwiązującego zagadkę logiczną, Piętnastkę, poprzez wykorzystanie różnych metod przeszukiwania stanu oraz przebadanie ich.

### 2. Wprowadzenie

Piętnastka to układanka logiczna składająca się z piętnastu klocków, numerowanych od 1 do 15, ułożonych na kwadratowej planszy, o wymiarach 4x4. Celem układanki jest przestawienie tak klocków, żeby ustawić je w kolejności rosnącej, przy czym pusty element musi znaleźć się na końcu. Przesuwanie klocków umożliwia nam puste miejsce na planszy.

Poszukiwanie rozwiązania łamigłówki jest zbliżone do znajdowania ścieżki w grafie, gdzie stan układanki jest węzłem w grafu.

W celu odnalezienia właściwej ścieżki stosujemy następujące strategie:

- BFS - Breadth First Search - przeszukiwanie "wszerz" rozpoczynając od zadanego wierzchołka w grafie, odwiedza wszystkie osiągalne z niego wierzchołki na tym samym poziomie rekursji, według ustalonej wcześniej kolejności sprawdzania. Następnie odwiedza wszystkie osiągalne wierzchołki pochodne, na kolejnym poziomie rekursji
- DFS - Depth First Search - przeszukiwanie "w głąb" rozpoczyna przechodzenie grafu od zadanego wierzchołka, odwiedzając pierwszy z pochodnych wierzchołków, kolejność przechodzenia jest wcześniej ustalana,

powtarzając to dla każdego pochodnego wierzchołka. Jeżeli algorytm nie będzie mógł wchodzić dalej (osiągnie zadaną maksymalną głębokość rekursji), cofa się o jeden poziom rekursji i bada krawędź kolejnego nieodwiedzonego jeszcze wierzchołka.

- A\* - Algorytm heurystyczny znajduje najkrótszą możliwą ścieżkę, jeśli taka istnieje. W przypadku piętnastki algorytm A\* tworzy ścieżkę wybierając wierzchołek tak, aby minimalizować wartość heurystyki. Metody obliczania tej wartości to: metoda Hamminga, gdzie obliczamy ile klocków znajduje się na niewłaściwych pozycjach, metoda Manhattan, gdzie liczymy jakie odległości dzielą klocki od ich docelowych miejsc.

### 3. Opis implementacji

Program został napisany w języku Python. Klasa Fifteen jest klasą główną natomiast plik FileManager.py odpowiada za wczytywanie układanki, zapis statystyk oraz zapis ułożonej układanki. W klasie Fifteen znajdują się wszystkie strategie oraz niezbędne operacje na układance, np. odnalezienie pozycji pustej, zamiana elementów, przedstawienie możliwych ruchów.

### 4. Materiały i metody

W tym miejscu należy opisać, jak przeprowadzone zostały wszystkie badania, których wyniki i dyskusja zamieszczane są w dalszych sekcjach. Opis ten powinien być na tyle dokładny, aby osoba czytająca go potrafiła wszystkie przeprowadzone badania samodzielnie powtórzyć w celu zweryfikowania ich poprawności. Przy opisie należy odwoływać się i stosować do opisanych w sekcji drugiej wzorów i oznaczeń, a także w jasny sposób opisać cel konkretnego testu. Najlepiej byłoby wyraźnie wyszczególnić (ponumerować) poszczególne eksperymenty tak, aby łatwo było się do nich odwoływać dalej.

### 5. Wyniki

#### BFS

##### 5.1. DRUL

DRUL
------

### 6. Dyskusja

Sekcja ta powinna zawierać dokładną interpretację uzyskanych wyników eksperymentów wraz ze szczegółowymi wnioskami z nich płynącymi. Najcenniejsze są, rzecz jasna, wnioski o charakterze uniwersalnym, które mogą być istotne przy innych, podobnych zadaniach. Należy również omówić i wyjaśnić wszystkie napotkane problemy (jeśli takie były). Każdy wniosek powinien mieć poparcie we wcześniej przeprowadzonych eksperymentach (odwołania

do konkretnych wyników). Jest to jedna z najważniejszych sekcji tego sprawozdania, gdyż prezentuje poziom zrozumienia badanego problemu.

## 7. Wnioski

W tej, przedostatniej, sekcji należy zamieścić podsumowanie najważniejszych wniosków z sekcji poprzedniej. Najlepiej jest je po prostu wypunktować. Znow, tak jak poprzednio, najistotniejsze są wnioski o charakterze uniwersalnym.

## Literatura

- [1] T. Oetiker, H. Partl, I. Hyna, E. Schlegl. *Nie za krótkie wprowadzenie do systemu  $\text{\LaTeX}2\epsilon$* , 2007, dostępny online.