Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 2 з дисципліни

«Проєктування алгоритмів»

«Неінформативний, інформативний та локальний пошук»

Варіант 3

Виконав студент ІП-14 Гайдучек Максим Андрійович

(шифр, прізвище, ім'я, по батькові)

Перевірив Соколовський Владислав Володимирович

( прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота №2**

**Тема роботи:** Проектування і аналіз алгоритмів зовнішнього сортування

**Мета роботи:** розглянути та дослідити алгоритми неінформативного, інформативного та локального пошуку. Провести порівняльний аналіз ефективності використання алгоритмів.

**Варіант:** 3

**Завдання**: Розв’язування лабіринту, алгоритми BFS та A\* з Мангетенською відстанню.

**Хід роботи**

* 1. Псевдокод алгоритму BFS

**ВВІД** maze, start, end;

start.visited = true

queue = [start]

**ПОКИ** queue != []

cell = queue.pop()

**ЯКЩО** cell == end

**ВИВІД** cell.path

**КІНЕЦЬ ЯКЩО**

**ВІД** 0 **ДО** 3 **З** i

x = cell.x + направлення\_x(i)

y = cell.y + направлення\_y(i)

**ЯКЩО** в\_межах(maze, x, y) **ТА** !cell.visited **ТА** cell.type == “СТЕЖКА”

neighbourCell = maze.getCell(x, y)

neighbourCell.visited = true

neighbourCell.path.addAll(cell.path)

neighbourCell.path.add(cell)

queue.push(neighbourCell)

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ВІД-ДО**

**КІНЕЦЬ ПОКИ**

* 1. Псевдокод алгоритму A\* з Мангетенською відстанню

*solving*:

**ВВІД** maze, start, end;

start.visited = true

queue = [start]

**ПОКИ** queue != []

cell = queue.pop()

**ЯКЩО** cell == end

**ВИВІД** cell.path

**КІНЕЦЬ ЯКЩО**

*calcManhattanDistance*(cell, end)

**ВІД** 0 **ДО** 3 **З** i

x = cell.x + направлення\_x(i)

y = cell.y + направлення\_y(i)

**ЯКЩО** в\_межах(maze, x, y) **ТА** !cell.visited **ТА** cell.type == “СТЕЖКА”

neighbourCell = maze.getCell(x, y)

neighbourCell.visited = true

neighbourCell.path.addAll(cell.path)

neighbourCell.path.add(cell)

*calcManhattanDistance*(cell, end)

*addToQueue*(neighbourCell)

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ВІД-ДО**

**КІНЕЦЬ ПОКИ**

*calcManhattanDistance*:

**ВВІД** cell, end

cell.distance = abs(cell.x – end.x) + abs(cell.y – end.y)

*addToQueue*:

**ВВІД** cell, queue

index = 0

item = queue[0]

**ПОКИ** index < queue.size – 1 ТА cell.distance + cell.path.length > item.distance + item.path.length

item = queue.(index + 1)

index = index + 1

**КІНЕЦЬ** **ПОКИ**

queue.insert(cell, index)

* 1. Код алгоритму BFS

Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

* 1. Код алгоритму A\* з Манхетенською відстанню

Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

1. Приклади роботи алгоритмів

На рисунках 3.1 і 3.2 показані приклади роботи програми для різних алгоритмів пошуку.

Изображение выглядит как текст, внешний

Автоматически созданное описание

Рисунок 3.1 – Алгоритм BFS

Изображение выглядит как текст, внешний

Автоматически созданное описание

Рисунок 3.2 – Алгоритм A\* з Мангетенською відстанню

* 1. Дослідження алгоритмів

В таблиці 4.1 наведені характеристики оцінювання алгоритму BFS, задачі розв’язування лабіринту для 20 початкових станів.

Таблиця 4.1 – Характеристики оцінювання алгоритму BFS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Початкові стани | Ітерації | Кількість глухих кутів | Всього станів | Всього станів у пам’яті |
| Лабіринт 21х21 | 174 | 518 | 692 | 174 |
| Лабіринт 25х25 | 249 | 737 | 992 | 255 |
| Лабіринт 29x29 | 394 | 1176 | 1572 | 396 |
| Лабіринт 33x33 | 470 | 1402 | 1876 | 474 |

Продовження таблиці 4.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Початкові стани | Ітерації | Кількість глухих кутів | Всього станів | Всього станів у пам’яті |
| Лабіринт 37x37 | 594 | 1774 | 2372 | 598 |
| Лабіринт 41x41 | 569 | 1696 | 2272 | 576 |
| Лабіринт 45x45 | 803 | 2398 | 3208 | 810 |
| Лабіринт 49x49 | 1131 | 3386 | 4520 | 1134 |
| Лабіринт 53x53 | 1147 | 3431 | 4584 | 1153 |
| Лабіринт 57x57 | 984 | 2934 | 3932 | 998 |
| Лабіринт 61x61 | 1803 | 5405 | 7208 | 1803 |
| Лабіринт 65x65 | 1869 | 5598 | 7472 | 1874 |
| Лабіринт 69x69 | 2188 | 6554 | 8748 | 2194 |
| Лабіринт 73x73 | 1971 | 5894 | 7880 | 1986 |
| Лабіринт 73x73 | 1971 | 5894 | 7880 | 1986 |
| Лабіринт 77x77 | 2271 | 6796 | 9080 | 2284 |
| Лабіринт 81x81 | 2737 | 8194 | 10944 | 2750 |
| Лабіринт 85x85 | 3085 | 9240 | 12336 | 3096 |
| Лабіринт 89x89 | 2985 | 8941 | 11936 | 2995 |
| Лабіринт 93x93 | 4055 | 12151 | 16216 | 4065 |
| Лабіринт 99x99 | 4203 | 12595 | 16808 | 4213 |

В таблиці 4.2 наведені характеристики оцінювання алгоритму A\* з Мангетенською відстанню, задачі розв’язування лабіринту для 20 початкових станів.

Таблиця 4.2 – Характеристики оцінювання алгоритму A\* з Мангетенською відстанню

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Початкові стани | Ітерації | Кількість глухих кутів | Всього станів | Всього станів у пам’яті |
| Лабіринт 21х21 | 109 | 317 | 432 | 115 |
| Лабіринт 25x25 | 145 | 422 | 576 | 154 |
| Лабіринт 29x29 | 293 | 868 | 1168 | 300 |
| Лабіринт 33x33 | 298 | 871 | 1188 | 317 |
| Лабіринт 37x37 | 401 | 1179 | 1600 | 421 |
| Лабіринт 41x41 | 385 | 1141 | 1536 | 395 |
| Лабіринт 45x45 | 405 | 1184 | 1616 | 432 |
| Лабіринт 49x49 | 817 | 2425 | 3264 | 839 |
| Лабіринт 53x53 | 713 | 2119 | 2848 | 729 |
| Лабіринт 57x57 | 426 | 1243 | 1700 | 457 |
| Лабіринт 61x61 | 1380 | 4116 | 5516 | 1400 |
| Лабіринт 65x65 | 1146 | 3418 | 4580 | 1162 |
| Лабіринт 69x69 | 1791 | 5341 | 7160 | 1819 |

Продовження таблиці 4.2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Початкові стани | Ітерації | Кількість глухих кутів | Всього станів | Всього станів у пам’яті |
| Лабіринт 73x73 | 1118 | 3329 | 4468 | 1139 |
| Лабіринт 77x77 | 1525 | 4542 | 6096 | 1554 |
| Лабіринт 81x81 | 1185 | 3515 | 4736 | 1221 |
| Лабіринт 85x85 | 2083 | 6226 | 8328 | 2102 |
| Лабіринт 89x89 | 2317 | 6936 | 9264 | 2328 |
| Лабіринт 93x93 | 2248 | 6711 | 8988 | 2277 |
| Лабіринт 99x99 | 3213 | 9620 | 12848 | 3228 |

**Висновок**: під час виконання даної лабораторної роботи було вирішено задачу проходження лабіринту алгоритмами BSF та A\* з розрахунком Мангетенською відстанню. Було розроблено псевдокод та код алгоритмів. Після опробування алгоритмів видно, що A\* виконує менше ітерацій та оброблює менше станів, чим BFS, а тому цей алгоритм виконує менше помилок (менше кількості глухих кутів) та являється більш оптимізованим.