Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

ЗВІТ

про виконання комп’ютерного практикуму № 2

з дисципліни

“Програмування систем штучного інтелекту”

Тема: Інформативний та неінформативний пошуки

|  |  |  |
| --- | --- | --- |
| Прийняв(ла): |  | Виконав: |
| Мажара О. О. |  | студент 5 го курсу  гр. ІП-92мп ФІОТ  Педоренко А.В. |

Київ 2019

**Постановка задачі**

1) Реалізувати програму, яка розв’язує поставлену задачу 8-ферзів за допомогою алгоритму локального пошуку HILL. Евристичну функцію обрати довільну для заданої задачі. Програму реалізувати на довільній мові процедурного програмування (C, C++, C#, Java, ython, Matlab, PHP, ...)

2) Провести 20 експериментів роботи алгоритму (для цього вміти генерувати випадковим чином початковий стан), які відрізняються характеристиками роботи алгоритму. За результатами експериментів визначити:

• середню кількість етапів (кроків), які знадобилось для досягнення розв’язку

• середню кількість випадків, коли алгоритм потрапляв в глухий кут (не міг знайти оптимальний розв’язок) – якщо таке можливе

• середній час роботи алгоритму в секундах

3) На додаткові бали. Порівняти результати розв’язання задачі за допомогою алгоритму локального пошуку із результати з лабораторної роботи No2.

**Результати**

Результати роботи алгоритму HILL на 20 експериметах:

Середня кількість кроків: 1505.3

Середню кількість випадків, коли алгоритм потрапляв в глухий кут: 0.1

Середній час пошуку рішення: 51 мс

**Висновок**

Оскільки для того, щоб бути більш впевненним, що алгоритм HILL спрацював правильно, його треба перезапустити досить велику кількість разів, він працює не дуже швидко, але так як одна ітерація коротка, повторення не так сильно впливають. Алгоритм використовує дуже мало пам’яті, оскільки сберігає тільки початковий, кращий результат та поточний стани. Алгоритм не завжди знаходить рішення.

Порівнюючи з алгоритмами інформативного та неінформативного пошуків, найважливіше те що локальний пошук не зберігає всі сгенеровані стани (чи весь фронт сгенерованих станів). Через велику кількість спроб алгоритм працює повільніше ніж A\*, але набагато швидше ніж BFS. Великим недоліком у порівнянні з глобальним пошуком є те, що локальний пошук не завжди знайде оптимальний результат.