

جامعة دمشق – كلية الهندسة المعلوماتية

قسم الذكاء الصناعي

عملي خوارزميات البحث الذكية

## مشروع عملي خوارزميات البحث الذكية

(جميع الأقسام)

الرجاء قراءة نص السؤال بالتفصيل، حيث أن كل جزئية فيه لها جزء من العلامة.

**المسألة الأولى:** إيجاد المسار الأمثل لمسألة تسليم الطرود لشركة توصيل طلبات

تقوم شركات توصيل الطرود مثل UPS و FedEx بتوصيل ملايين الطرود يومياً باستخدام آلاف المركبات. حيث أن عليهم أن يقرروا أي طرد يتم تحميله وأين ومتى... هذه مشكلة معقدة بسبب وجود جميع أنواع القيود مثل الوقت والتكلفة والقدرة...

في هذه المسألة تخيل نفسك أنك مالك لشركة توصيل طرود قد بدأت بالعمل حديثاً، وعليك إيجاد مخطط تسليم بسيط لشاحنة في شركة التوصيل.

**نمذجة المسألة:**

سنفترض أنه لدينا السيناريو التالي:

يمكن عرض المدينة التي تعيش فيها على أنها شبكة  $n \times m$ ، هي أبعاد المدينة مستطيلة الشكل، تكون كل خلية فيها إما مسار (.) أو مشغولة بمبنى (#)، ولديك شاحنة واحدة فقط (T).

سنفرض أيضاً أن لديك k طرداً تريد توصيله حيث يكون للطرد  $i$  موقع استلام  $(P_i)$  وموقع تسليم  $(D_i)$ .

فيما يلي مثال على ذلك:

$D_0$	.	.	.	.	$T$
.	#	#	$P_1$	.	.
.	.	#	#	.	.
$P_0$	.	.	.	.	$D_1$

هناك 6 عمليات يمكنك القيام بها أثناء التجوال<sup>1</sup>:

- 1- التحرك شمالاً (إلى أعلى)
- 2- التحرك جنوباً (إلى أسفل)
- 3- التحرك شرقاً (إلى اليمين)
- 4- التحرك غرباً (إلى اليسار)
- 5- استلام طرد.
- 6- تسليم طرد.

وأخيراً عليك العودة إلى موقع البداية، طبعاً لا يمكنك المرور خلال المباني، وأيضاً عند دخول شاحنتك للموقع  $P_i$  يمكنك أن تختار استلام الطرد  $i$  في حال لم يكن قد تم استلامه مسبقاً، وعند دخول شاحنتك للموقع  $D_i$  يمكنك أن تختار تسليم الطرد  $i$  في حال تم استلامه مسبقاً، انتبه أنه يمكنك المرور بالمواقع  $P_i$  و  $D_i$  بدون استلام أو تسليم أي طرد أي باعتبارها مسارات أيضاً، وأيضاً يمكنك المرور بالموقع نفسه أكثر من مرة.

كلفة الانتقال من خلية إلى خلية مجاورة هي 1، بالإضافة إلى عدد الطرود التي يتم نقلها في هذه اللحظة (باعتبارها أوزناً إضافية تحملها الشاحنة).

الهدف هو تسليم جميع الطرود والعودة إلى موقع البداية بأقل كلفة ممكنة.

**المطلوب:**

- 1- قم بتمثيل المسألة باعتبارها مسألة بحث state space search، ما الذي يمثل في مسألتك حالة (state)؟ ما هو تابع الانتقال؟ ما هي الكلفة؟ ما هي حالة البداية؟ وما هي حالة النهاية؟ حقق ذلك برمجياً باستخدام بنية تراها مناسبة.

<sup>1</sup> حافظ على ترتيب خطوات الحركة المعطى نفسه في نص السؤال

2- قم بحل المسألة باستخدام خوارزمية uniform cost search، لإيجاد الحل بالكلفة الأقل، قم بطباعة الخرج الممثل بمخطط التسليم النهائي، يمكنك التعبير عن الطريق الذي ستعبره الشاحنة بأرقام، بالإضافة إلى رموز خاصة تعبر عن استلام أو تسليم طرد في كل خطوة (أو بأي شكل آخر تراه مناسباً)، مثلاً يمكن أن يكون للحل الشكل التالي:

$15^d D_0$	16	17	2 18	1 19	$T$ 20
14	#	#	$3^p P_1$	4	.
13	.	#	#	5	.
$12^p P_0$	11	10	9	6 8	$7^d D_1$

بالإضافة إلى ذلك قم بطباعة كلفة الحل، وزمن التنفيذ وعدد العقد التي تم معالجتها (إخراجها من queue البحث).

3- لنقم بتسريع البحث بتطبيق خوارزمية  $A^*$ ، قم بتعريف مجموعة توابع تقوم بحساب تجريبية heuristic لحالة ما (تذكر أن التجريبية هي القيمة التقديرية الأصغرية للكلفة بين الحالة الحالية والهدف)<sup>2</sup> قم بطباعة الخرج السابق من أجل كل من التجريبيات التالية:

a. التجربة الأولى: افترض أن الشاحنة تريد العودة إلى موقع البداية (مهماً تسليم الطرود) أي أنك ستحسب الكلفة التقديرية لوصول شاحنتك للهدف من موقعها الحالي<sup>3</sup>.

b. التجربة الثانية: سنقوم بالتحسين في حساب الكلفة بشكل واقعي أكثر، حيث أن التجربة السابقة لا تكون حقيقة إلا في حالة كون الشاحنة قد سلمت جميع طرودها، لنفترض أن الشاحنة تريد تسليم طرد واحد فقط (وليكن الطرد<sup>4</sup>) احسب الكلفة التقديرية لتسليم الشاحنة لهذا الطرد – في حال لم تكن سلمته في الحالة التي يتم حساب التجربة لها – آخذاً بعين الاعتبار جميع المسافات التي يجب أن تقطعها

<sup>2</sup> التجريبيات التي سنقوم بتحقيقها سنقوم بحساب مسافات خط النظر بين مواقع مختلفة في الجدول بدون أن تأخذ بعين الاعتبار المباني، أي كأن يمكن لشاحنتك القيادة عبرها.

<sup>3</sup> أي أنك ستحسب مسافة خط النظر بين موقع الشاحنة الحالي وموقع البداية.

<sup>4</sup> في التجربة افترض أنها تريد تسليم الطرد 0 في حال لم تسلمه قبل ذلك

بحسب موقع الشاحنة وحالة الاستلام والتسليم لهذا الطرد في الحالة التي تحسب لها (أي هل قمت باستلام الطرد بعد، أو هل قمت بتسليمه...) وأيضاً كلفة وزن الطرد في حال كنت تحمل طرداً أو أكثر...

c. التجربة الثالثة: هنا سنحاول أن نجعل قيمة الكلفة المحسوبة في التجربة السابقة

قريبة أكثر من الواقع، سنفترض أنك الكلفة هي كلفة تسليم أكثر طرد مكلف<sup>5</sup>

4- قم بتجريب الطلبات 2 و 3 على كل من الدخول التالية (الأفضل قراءة الدخول من ملف نصي):

1.

```
. . . . # T
. P0 # # . # .
D0 . . # . . .
. . . # . . .
```

2.

```
. . . . # T
. . # # . # .
. . P0 # . . .
. . . . # D0
```

3.

```
. . # T # . .
. . # P0 # . .
. . # . # . .
D0 . . P1 . . D1
```

ثم ضع في جدول كلفة الحل، وزمن التنفيذ وعدد العقد التي تم معالجتها من أجل منها، هل وصلت جميعها لنفس الحلول؟

<sup>5</sup> يمكنك الاستعانة بالكود المكتوب في التجربة السابقة وحساب الكلفة من أجل جميع الطرود (وليس الطرد 0، وثم أخذ الكلفة الأكبر بينها).

## المسألة الثانية: لعبة Blob Wars



يمكنك لعب اللعبة وتجربتها على الرابط التالي: <https://www.twoplayergames.org/game/blob-wars>

خذ استراحة! وقم بلعب عدة أدوار من هذه اللعبة بعد قراءة التعليمات حتى تفهم قواعدها تماماً. والمطلوب منك برمجة هذه اللعبة بحيث يمكن لعبها بشكل تفاعلي مع الحاسب، مراعيًا إمكانية لعب هذه اللعبة باستخدام أحجام لوح مختلفة، وذلك باستخدام الخوارزميات التي استعرضناها في هذه المادة.

لا داعي لأن تعني بطريقة الإظهار، يكفي أن يكون الدخل والخروج واضحاً حتى لو كان بدون واجهة GUI.

قم بتعريف تابع يقيم حالة ما بحسب جودتها بالنسبة للاعب ما، لكي تتمكن من برمجة اللعبة بحيث تأخذ بعين الاعتبار عدد معين من الخطوات اللاحقة (عمق الشجرة) وبالتالي تتيح للمستخدم في بداية اللعبة تحديد المستوى الذي يرغب أن يلعب به، المستوى الأصعب يعني عمق شجرة أكبر.

### أمور تنظيمية:

- تسلم الوظيفة يوم الخميس 2022-1-6 وتتم المقابلات في نفس اليوم.
- عدد طلاب المجموعة الواحدة هو خمسة على الأكثر، يمكن لطلاب من فئات مختلفة ضمن القسم الواحد أن يشتركوا معاً.
- يسلم تقرير يحوي على الأفكار الأساسية وطريقة تمثيل المسألة وحلها، بالإضافة إلى النتائج ومقارنتها.
- لا داعي لتسليم CD يوم المقابلة، يسلم الكود والتقرير بصيغة pdf على فلاش مباشرة بعد وضعة بمجلد يحوي أسماء الطلاب وقسمهم.
- المطلوب من طلاب المجموعة التشارك جميعاً بحل المسألة، وهذا يعني أن الطلاب معاً يجتمعون لوضع البنية المناسبة لكل مسألة، وتحديد الصفوف والتوابع التي يجب تحقيقها، مع تحديد دخل وخرج كل منها، ثم تقسيم تنجيز (implementation) البنية والتوابع فيما بينهم، وبالتالي يكون جميع طلاب المجموعة على دراية كاملة بطريقة حل كل مسألة. هكذا يمكنكم جميعاً من وضع حلول أفضل للمسألة والتشارك في حل المشكلات، ويزيد من سرعة أداء المسألة.
- لضمان ذلك يطلب من طلاب كل مجموعة تصوير الورقة التي كتبوا عليها البنية البرمجية عندما اجتمعوا معاً وأيضاً تقسيم الكود الذي اتفقوا عليه (ولو كان هذا قد تغير بعد ذلك) مكتوباً على هذه الورقة، ووضع هذه الصورة في التقرير.
- يجب على كل طالب كتابة الجزء الذي قام بتنفيذه من المسائل في التقرير (عوضاً عن إيكال مهمة كتابة التقرير لأحد الأفراد) مع تحديد اسم الطالب الذي قام بتنفيذ هذا الجزء.
- علامة المشروع ليست واحدة لجميع أفراد المجموعة، في حال عدم اشتراك بعض الطلاب بحل الوظيفة (والاكتفاء بفهم الكود) ممكن أن ينالوا علامة الصفر عليها.
- أي وظيفة منقولة أو منقول منها ستنال علامة الصفر.



Good  
Luck!