كلية الهندسة

قسم الهندسة المعلوماتية

مقرر خوارزميات بحث ذكية

د. غزوان علي ريّا

محاضرات الأسبوع السابع

الفصل الأول 2024–2025

البحث المحلي Local Search

خوارزمية تسلق الهضبة Hill Climbing Algorithm

تسلق الهضبة البسيط Simple Hill Climbing

تسلق الهضبة عبر الصعود الأكثر انحداراً Steepest Ascent Hill Climbing

تسلق الهضبة العشوائي Stochastic Hill Climbing

تسلق الهضبة المعاود للبدء بطريقة عشوائية Random-Restart Hill Climbing

البحث الشعاعي/البحث عبر الحزم Beam Search

البحث المحلي Local Search:

تم تصميم خوارزميات البحث التي رأيناها حتى الآن لاستكشاف طرق البحث بشكل منهجي. يتم تحقيق هذه المنهجية من خلال الاحتفاظ بمسار واحد أو أكثر في الذاكرة وذلك عن طريق تسجيل الحالات التي يتم استكشافها على طول المسار.

ولكن وفي العديد من المسائل، يكون الطريق إلى الهدف غير مهم. على سبيل المثال، في مشكلة n-queens ما يهم هو التكوين النهائي للملكات، وليس الترتيب المتبع.

ينطبق ذلك على العديد من التطبيقات مثل تصميم الدارات المتكاملة، وتخطيط أرضية المصنع، والاتصالات السلكية واللاسلكية، وتحسين الشبكة، وتوجيه المركبات.

إذا كان الطريق إلى الهدف غير مهم، فقد نفكر في فئة مختلفة من الخوارزميات وهي Local Search Algorithms تلك التي لا تبالي بشأن المسارات على الإطلاق.

تعمل خوارزميات البحث المحلية باستخدام العقدة الحالية – عقدة حالية واحدة– (بدلاً من مسارات) وتهتم فقط بالجار الذي ستنتقل إليه من تلك العقدة. عادة لا يتم الاحتفاظ بالمسارات التي يتبعها البحث.

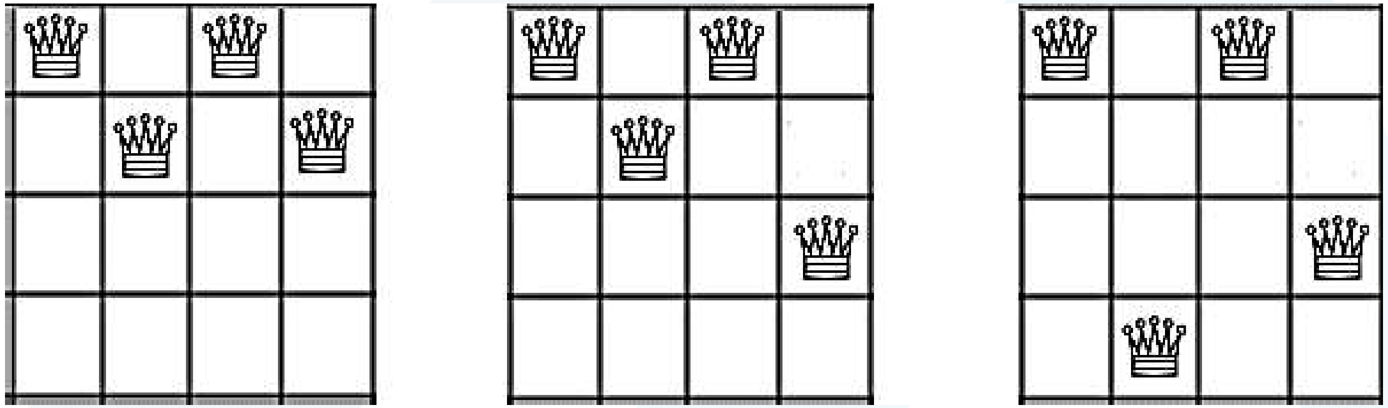
على الرغم من أنها محلية وليست منهجية، إلا أنه لها **ميزتان رئيسيتان:**

تستخدم ذاكرة قليلة جداً، كما يمكنها إيجاد حلول معقولة لبعض المسائل التي تكون فيها الخوارزميات المنهجية غير مناسبة.

بالإضافة إلى العثور على الأهداف، تعد خوارزميات البحث المحلية مفيدة لحل مشكلات Optimization والتي يكون الهدف منها هو العثور على أفضل حل لمشكلة ما فالعديد من مشاكل التحسين لا تتناسب مع نموذج البحث "القياسي" الذي تم شرحه مسبقاً.

في هذه الخوارزميات ليس من الضروري إيجاد الطريق الأفضلي للهدف فتكاليف الوصلات غير مهمة، هذه الخوارزميات أيضاً مفيدة في حال كان فضاء البحث كبيراً.

مثلاً في مسألة N-Queens والتي نهدف فيها لوضع مجموعة من الملكات عددها N على رقعة بحجم N×N دون أن تستطيع ملكة (الوزير في الشطرنج) القضاء على ملكة أخرى (أي ليس هناك ملكتان على نفس السطر أو العمود أو القطر) تعطي خوارزميات البحث المحلي حلاً مقبولاً وتوضح الصورة الآتية بعض الحالات الممكنة ل N-Queens.



خوارزمية تسلق الهضبة Hill Climbing Algorithm:

تملك عدة نسخ مثل تسلق الهضبة البسيط Simple Hill Climbing، تسلق الهضبة عبر الصعود الأكثر انحداراً Steepest Ascent Hill Climbing وغيرها.

تحاول هذه الخوارزميات إيجاد حلول جيدة لمسائل إيجاد الحل الأفضلي Optimization Problems.

يعد تسلق التل أسلوب للتحسين الرياضي Optimization ينتمي إلى عائلة البحث المحلي Local Search حيث أنها خوارزمية متكررة **تبدأ بحل تعسفي لمشكلة ما ثم تحاول إيجاد حل أفضل عن طريق إجراء تغيير تدريجي على الحل**.

خوارزمية تسلق الهضبة لا ترى أبعد من الجيران المباشرين للحالة الحالية، وتهتم فقط بأفضل عقدة مجاورة لتوسعتها، ويتم اختيار أفضل عقدة مجاورة عبر تابع التقدير Evaluation Function، بينما خوارزمية البحث الأفضل أولاً Best-First Search تنظر أبعد من الجيران المباشرين لتجد أقربهم للهدف (عبر التابع الحدسي) وثم تكمل مع أفضلهم.

يمكن تشبيه هذه الخوارزمية بالحالة التالية:

تخيل نفسك تتسلق جبل ايفيريست محاطاً بضباب كثيف وأنت تعاني من فقد للذاكرة، فتتسلق وتصعد بينما تركز على الخطوات المجاورة التي تقوم باتخاذها، طبعاً يجب أن تكون حذراً في مثل هذه الحالات، فيمكن أن تصل لنقطة إن عبرتها ستعود للأسفل، في النهاية تظن نفسك أنك على قمة الجبل ويتلاشى الضباب فتكتشف أن قمة ايفيريست مازالت بعيدة جداً، يمكن تشبيه خوارزمية تسلق الهضبة بمثل هذا الموقف.

لنفهم مفهوم خوارزمية تسلق الهضة، سنعرض الشكل الآتي الذي يمثل قمة (الحالة الهدف) وحالة المتسلق الحالية، فالمناطق الطبوغرافية في الشكل يمكن تعريفها بالشكل الآتي:

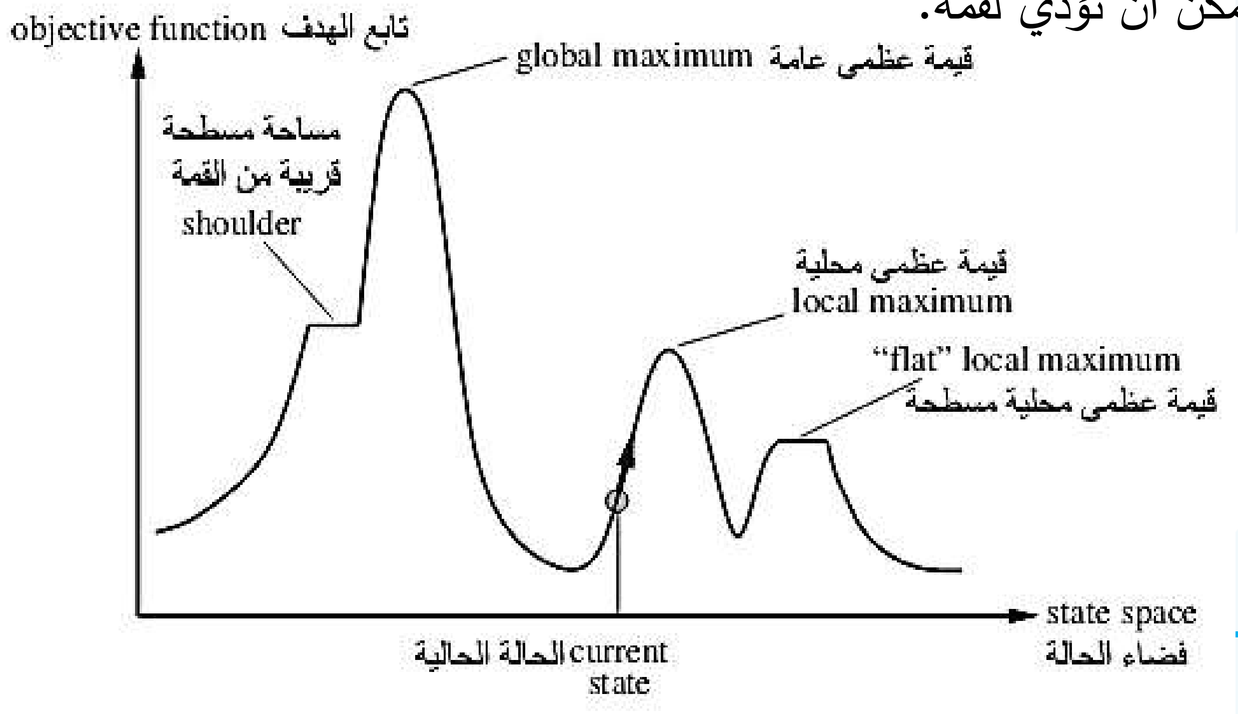
يمثل الشكل الآتي شكلاً متعلقاً بفضاء حالة أحادي البعد حيث يقابل الارتفاع فيه قيمة تابع الهدف.

القيمة العظمى العامة Global Maximum: هي أعلى نقطة في الهضبة والتي هي الحالة الهدف.

القيمة العظمى المحلية Local Maximum: هي نقطة أعلى من بقية النقاط ولكن أصغر من القيمة العظمى العامة.

القيمة العظمى المحلية المسطحة Flat Local Maximum: هي نقطة مسطحة من الهضبة لا تملك حافة صاعدة أو حافة هابطة، هي منطقة مشبعة من الهضبة.

المساحة المسطحة القريبة من القمة Shoulder: هي منطقة مسطحة يمكن أن تؤدي لقمة.



أنواع خوارزميات تسلق الهضبة:

كما ذكرنا سابقاً هناك تسلق الهضبة البسيط Simple Hill Climbing،

تسلق الهضبة عبر الصعود الأكثر انحداراً Steepest Ascent Hill Climbing، سنذكر أيضاً

تسلق الهضبة العشوائي Stochastic Hill Climbing،

تسلق الهضبة المعاود للبدء بطريقة عشوائية Random-Restart Hill Climbing.

<u>تسلق الهضبة البسيط Simple Hill Climbing:</u>

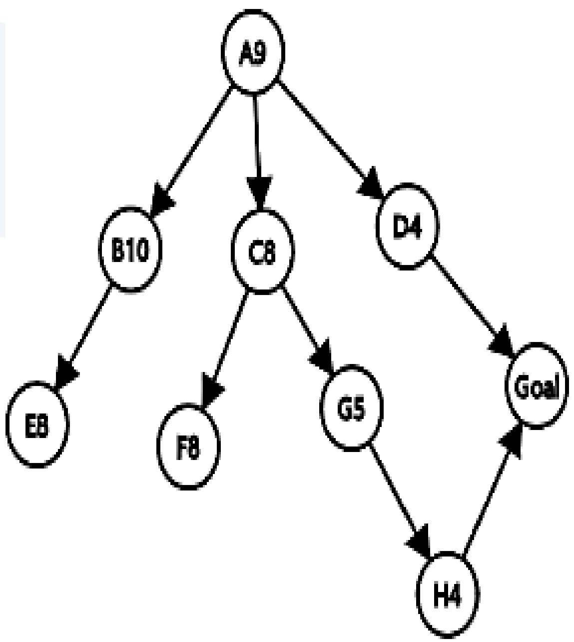
هو نمط بسيط لا يستعمل بنى معطيات معقدة، فهو يتعامل مع الذاكرة بكفء Efficient on memory، لا يسمح هذا النمط بالتراجع no backtracking، العقد التي لا يتم اختيارها خلال البحث يتم التخلص منها فورياً، يقوم ببحث محلي يختار فيه أول حالة أفضل من العقدة الحالية، ليس من الضروري أن يجد هدفاً (**فهو ليس تاماً not complete**)، ليس من الضروري أن يجد الطريق الأفضلي (**فهو ليس أفضلياً not optimal**)، يمكن أن يعلق في القيم المحلية الكبرى Local Maxima أو القيم المحلية الصغرى Local Minima،

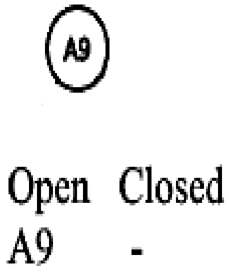
يعمل بشكل جيد مع تابع حدسي دقيق.

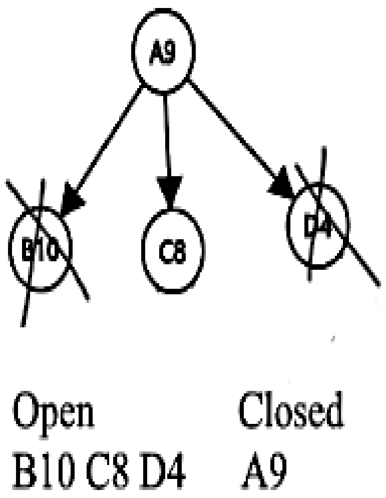
فخوارزمية تسلق الهضبة البسيط هي خوارزمية تساعد على تسلق جبل في فضاء ثنائي البعد، في كل خطوة يرى المتسلق الخطوة التالية ويقرر إن كان سيبقى أو يتحرك، وعندما يصل لأعلى نقطة يظن أنه في قمة الهضبة أو القيمة العظمى العامة Global Maximum مع أنه من المحتمل أنه وصل لنهاية محلية عظمى Local Maximum.

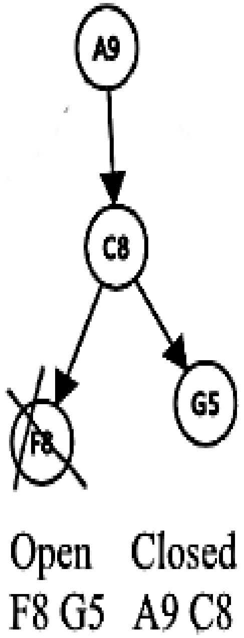
خوارزمية تسلق الهضبة البسيط هي بالشكل التالي:

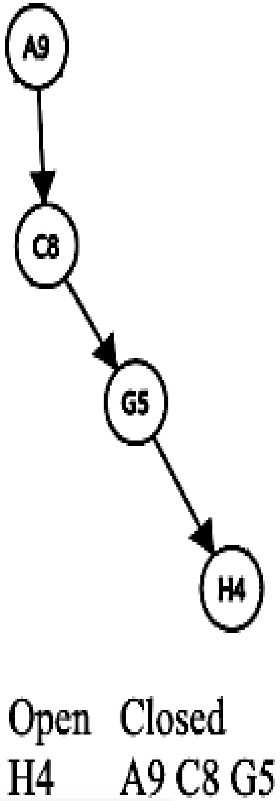
```
Create a CURRENT node, NEIGHBOUR node,
and
a GOAL node.
If the CURRENT node=GOAL node, return
GOAL and
terminate the search.
Else CURRENT node < NEIGHBOUR node, move
ahead.
Loop until the goal is not reached or a
point is not found.
```

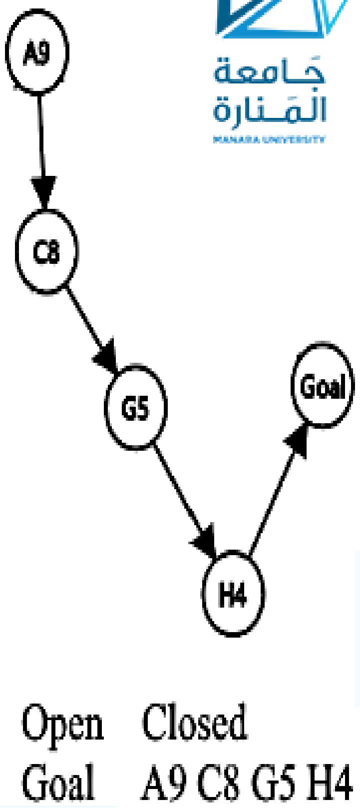












Open	Closed
A_9	-
B_{10} C_8 D_4	A_9
F_8 G_5	A_9 C_8
H_4	A_9 C_8 G_5
Goal	A_9 C_8 G_5 H_4
Stop	A_9 C_8 G_5 H_4 Goal

نقوم في أول مرحلة بإضافة عقدة البداية A وتوسعتها (إضافة أبنائها)

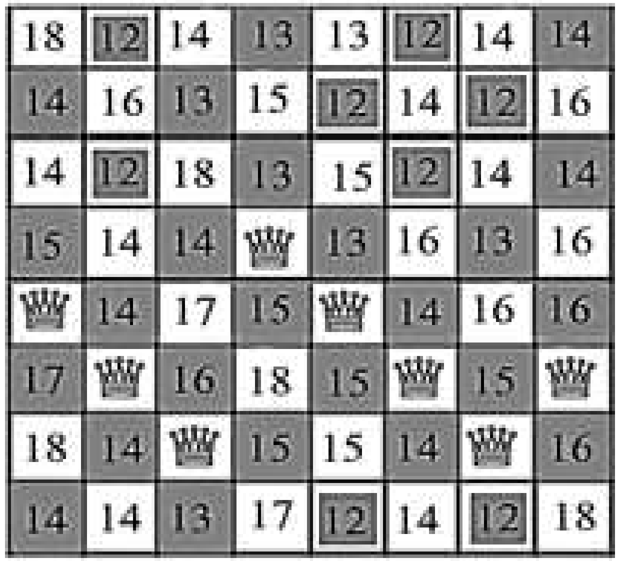
نختبر قيمة التابع الحدسي لأول ابن وهو العقدة B فنلاحظ أنها أكبر من قيمة التابع الحدسي للأب A (10>8) فننتقل للعقدة التالية وهي C نلاحظ أن قيمة التابع الحدسي لها أصغر من قيمة التابع الحدسي ل A فنختارها ونحذف بقية أبناء A (أشرنا للعقد المحذوفة)

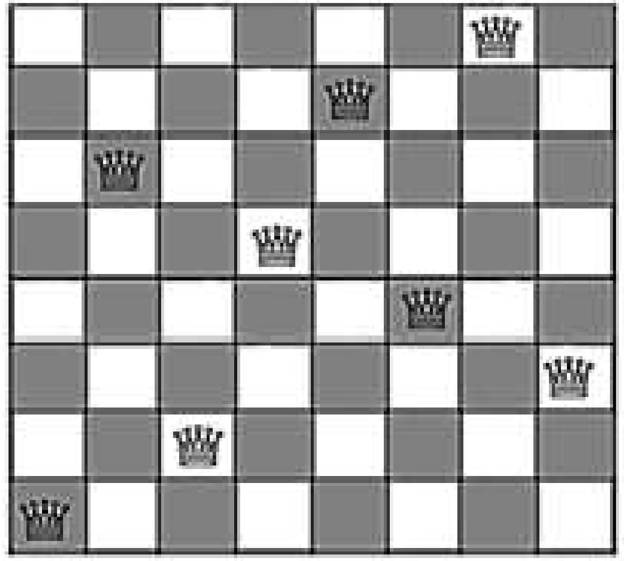
ثم نقوم بتوسعة العقدة المختارة C فنختبر العقدة F نلاحظ أن قيمتها مساوية لقيمة التابع الحدسي ل C فننتقل للعقدة التالية G نلاحظ أن قيمة التابع الحدسي لها أصغر من قيمة التابع الحدسي ل C فنختارها ونحذف بقية العقد أبناء C

ثم نكرر الخطوات حتى نصل للعقدة الهدف، فيكون الحل هو A C G H Goal مع ملاحظة أن هناك بشكل واضح طريق أقصر هو A D Goal.

أي ببساطة في كل خطوة نختبر العقد (بشكل عشوائي) حتى نصل لعقدة تكون قيمة التابع الحدسي لها أصغر من قيمة التابع الحدسي لأبيها فنختارها ونحذف بقية العقد ونكرر حتى نصل للهدف أو نصل لعقدة كل أبناءها يملكون قيمة تابع حدسي أكبر أو تساويها فهنا <u>**تتوقف الخوارزمية دون إيجاد الحل.**</u>

في مسألة الملكات عندما يكون هناك 8 ملكات وبدأنا بتابع حدسي قيمته 17 (يمثل هذا التابع عدد التضاربات بين الملكات) توضح الصورة اليسارية كيف يمكن للتابع الحدسي أن يتغير عبر تحريك الملكات في العمود وأفضل الحركات مشار لها بمربع، توضح الصورة اليمينية قيمة صغرى محلية يملك التابع الحدسي فيها قيمة 1 ولكن أي تحريك يؤدي لحالة جديدة تملك قيمة تابع حدسي أكبر.





تسلق الهضبة عبر الصعود الأكثر انحداراً Steepest Ascent Hill Climbing:

يختلف هذا النمط عن تسلق الهضبة البسيط، فخلافاً لتسلق الهضبة البسيط، يأخذ كل العقد التالية بالحسبان،

ثم يقارنها ويختار العقدة الأقرب للحل، ويشبه البحث الأفضل أولاً Best-First Search لأنه يركز على كل عقدة بدلاً من عقدة واحدة.

فخوارزمية تسلق الهضبة عبر الصعود الأكثر انحداراً تساعدك على تصور الجبل في فضاء ثلاثي الأبعاد، وبالتالي في كل خطوة يمكن أن يكون هناك عدة خطوات متتالية فنقوم باختيار أكبرها، لأنه في هذه الحالة من الممكن أن يقود الطريق المتسلق لأعلى الهضبة (نهاية عظمى عامة).

ففي خوارزمية تسلق الهضبة البسيط تبحث الخوارزمية عن أي قيمة تالية تقربها إلى الهدف ويمكن أن تنتهي بقيم محلية صغرى أو قيم محلية عظمى، بينما تبحث خوارزمية تسلق الهضبة عبر الصعود الأكثر انحداراً تبحث عن أفضل قيمة تالية تقربها من الهدف.

ولكن كلا الخوارزميتين لديهما حدود فيعلقان في قيمة محلية كبرى ويمكن معالجة هذا الأمر عبر التخمير المُحاكى Simulated Annealing.

تفشل كلا الخوارزميتين عندما لا يكون هناك عقدة أكثر قرباً للهدف.

خوارزمية تسلق الهضبة عبر الصعود الأكثر انحداراً هي كما يلي:

Create a **CURRENT** node and a **GOAL** node.

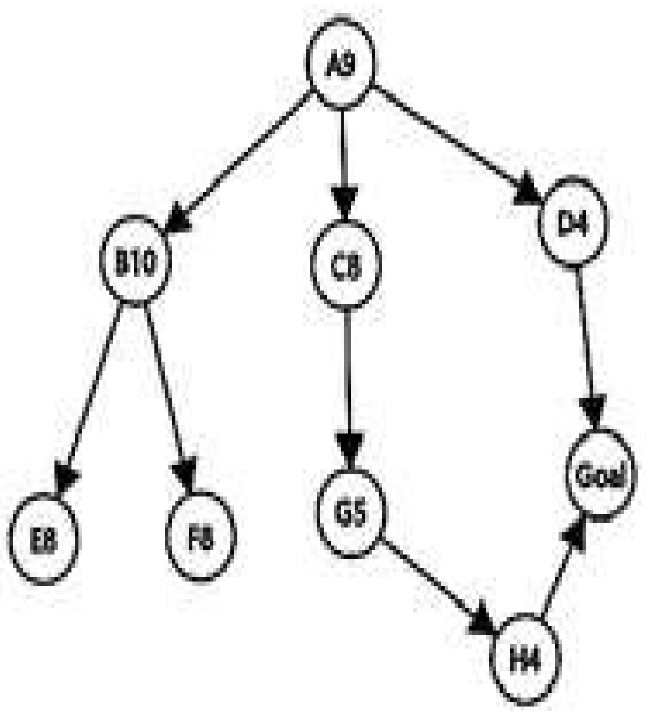
If the **CURRENT node=GOAL** node, return **GOAL** and terminate the search.

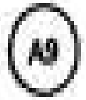
Loop until a better node is not found to reach the solution.

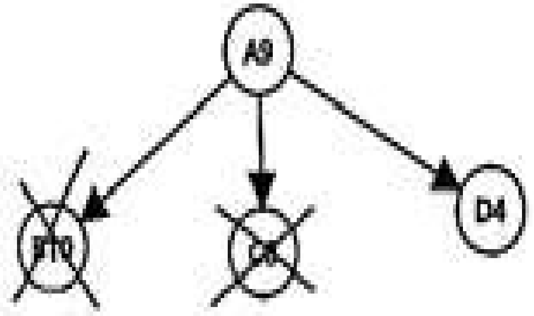
If there is any better successor node present, expand it.

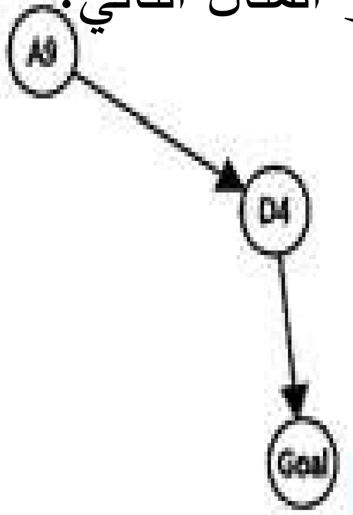
When the **GOAL** is attained, return **GOAL** and terminate.

سنقوم بشرح الخوارزمية عبر المثال التالي:









Open	Closed
A_9	-
B_{10} C_8 D_4	A_9
Goal	A_9 D_4
Stop	A_9 D_4 Goal

تشبه هذه الخوارزمية خوارزمية تسلق الهضبة البسيط إلا أننا هنا نمر على كل الأبناء ونختار أفضل عقدة (في الخطوة الثانية اخترنا C لأن لها أقل قيمة تابع حدسي) ونكرر الخطوات حتى نصل للعقدة الهدف.

تسلق الهضبة العشوائي Stochastic Hill Climbing:

كما ذكرنا سابقاً، تم اختراع عدة نسخ من خوارزميات تسلق الهضبة، أحدها تسلق الهضبة العشوائي، والتي يتم فيها اختيار قيمة عشوائية من بين قيم الحافة الصاعدة، وهذه غالباً ما تتقارب بشكل أبطأ من تسلق الهضبة عبر الصعود الأكثر انحداراً، ولكن في بعض الأشكال لفضاءات الحالة، تعطي نتائج أفضل.

خوارزميات تسلق الهضبة التي وُصِفت حتى الآن هي غير تامة Incomplete حيث أنها غالباً تفشل في إيجاد هدف وغالباً ما تعلق في القيمة الكبرى المحلية.

تسلق الهضبة المعاود للبدء بطريقة عشوائية Random-Restart Hill Climbing:

يعتمد على المقولة الشهيرة "**إن لم تنجح في البداية فقم بإعادة المحاولة مرة أخرى**" فيقوم بسلسلة من عمليات بحث تسلق الهضبة انطلاقاً من مجموعة حالات بدائية عشوائية (توليد حالة عشوائية من فضاء حالات معرف بشكل صريح هو مسألة صعبة بذاته) حتى يتواجد الهدف.

هو تام بشكل بديهي واحتمالية نجاحه تقترب من ال 1، لأنه في النهاية سيولد حالة هدف كحالة بدائية.

إن كان لكل بحث تسلق هضبة احتمالية p للنجاح، فعدد إعادات المحاولة المتوقعة هو $\frac{1}{p}$، فبالنسبة لمسألة الملكات عندما تكون هناك 8 ملكات من دون حركات جانبية مسموحة يكون احتمال النجاح $p \approx 0.14$، ولهذا نحتاج 7.14 أي تقريباً 7 تكرارات لنجد الهدف (6 من هذه التكرارات ستفشل وواحدة ستنجح).

ففي حالة مسألة ال 8 ملكات تسلق الهضبة المعاود للبدء بطريقة عشوائية فعال جداً، لأنه حتى مع 3 مليون ملكة تستطيع هذه الطريقة إيجاد حل في أقل من دقيقة.

يعتمد نجاح تسلق التل بشكل كبير على شكل فضاء الحالات، فإن كان هناك القليل من القيم المحلية الكبرى ستجد هذه الخوارزمية حلاً جيداً بسرعة.

أما مسائل NP-hard غالباً لديها عدد أسي من القيم المحلية الكبرى تعلق بها، ولكن بالرغم من هذا، غالباً ما يمكن إيجاد قيمة محلية عظمى جيدة بشكل معقول بعدد قليل من عمليات معاودة البدء.

البحث الشعاعي/البحث عبر الحزم Beam Search:

هو خوارزمية حدسية تكتشف البيان عبر توسعة أكثر عقدة واعدة في مجموعة محدودة، وهو تحسين للبحث الأفضل أولاً Best-First Search يقلل من احتياجات الذاكرة، فالبحث الأفضل أولاً هو بحث يرتب كل الحلول الجزئية وفق تابع حدسي ما، ولكن في البحث الشعاعي، **فقط عدد محدد مسبقاً من أفضل الحلول الجزئية الممكنة يتم الاحتفاظ بها كحلول مُرشَحة**.

يستخدم البحث الأفضل أولاً Best-First Search لبناء شجرة بحثه، وفي كل مستوى من الشجرة يولد كل الخلف للحالات في المستوى الحالي، ويرتبهم حسب ترتيب تصاعدي لقيمة التابع الحدسي، فيخزن عدد محدد مسبقاً β من أفضل الحالات في كل مستوى تسمى **بحجم الشعاع** أو عرض الحزمة beamwidth، وفقط هذه الحالات يتم توسيعها في الخطوات التالية.

كلما كبر حجم الشعاع تم تقليم (حذف) حالات أقل، وعندما يكون حجم الشعاع غير منته يصبح البحث الشعاعي مطابقاً للبحث الأفضل أولاً Best-First Search. فيقوم حجم الشعاع بالحد من الذاكرة اللازمة لتنفيذ البحث، وبما أن حالة الهدف يمكن أن تصبح محذوفة (عبر تقليمها)، **فيضحّي البحث الشعاعي بالتمام** Completeness.

البحث الشعاعي غير أفضلي Not Optimal أي ليس أكيداً أنه يسجل الحل الأفضل.

فبالعموم يرد البحث الشعاعي أول حل موجود، وعندما يصل لأكبر مستوى مسموح بالبحث ستقوم الخوارزمية بتقدير الحلول التي تم إيجادها خلال البحث في أعماق مختلفة ويرد أفضلها الذي يملك أكبر احتمال، أيضاً يمكن أن يكون حجم الشعاع ثابتاً Fixed أو متغيراً Variable، فأحد الطرق تستعمل حجم شعاع متغير في البداية كحد أدنى، إن لم يتواجد حل فيتم توسيع حجم الشعاع وتكرر العملية.

عناصر البحث الشعاعي:

يأخذ البحث الشعاعي ثلاثة عناصر كدخل له:

المسألة: وغالباً ما تُمثل كبيان وتحوي مجموعة من العقد فيها عقدة أو أكثر تمثل الهدف.

مجموعة من القواعد الحدسية لتقليم الشجرة: هي قواعد محددة حسب نطاق المسألة وتقوم بتقليم العقد غير الواعدة unfavorable nodes من الذاكرة المتعلقة بنطاق المسألة.

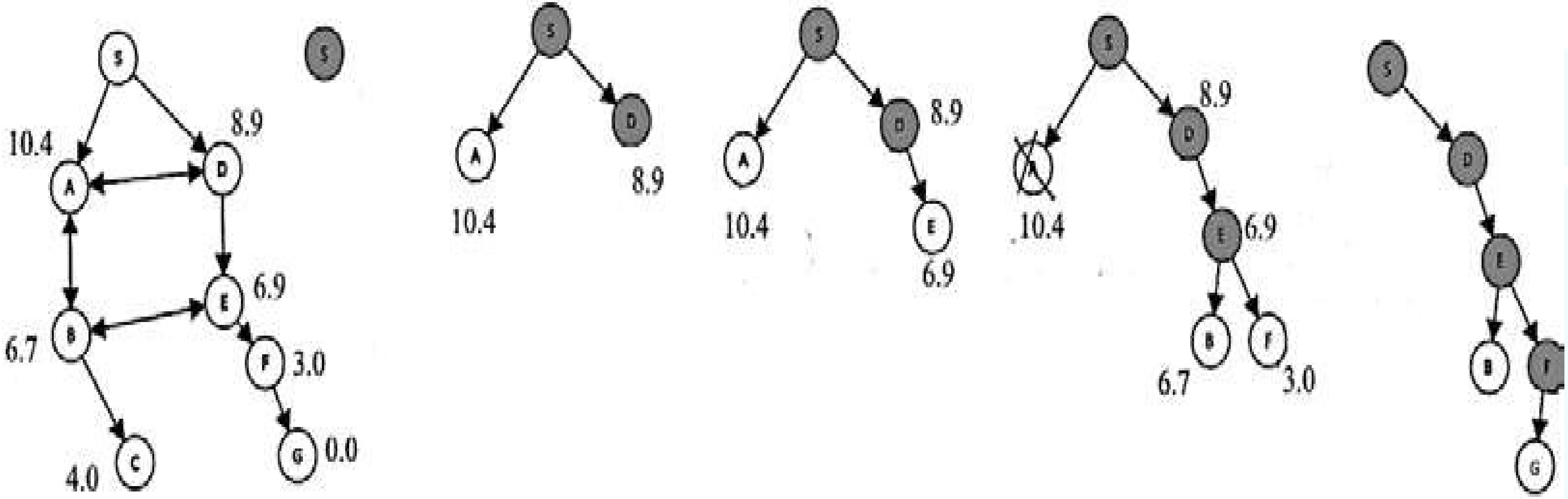
ذاكرة مع سعة محدودة: هي الذاكرة حيث يتم تخزين الشعاع، عندما تصبح الذاكرة ممتلئة ويتم إضافة عقدة للشعاع، فسيتم مسح أكثر العقد تكلفة، بحيث لا يتم تجاوز حد الذاكرة.

وخوارزميته هي بالشكل التالي:

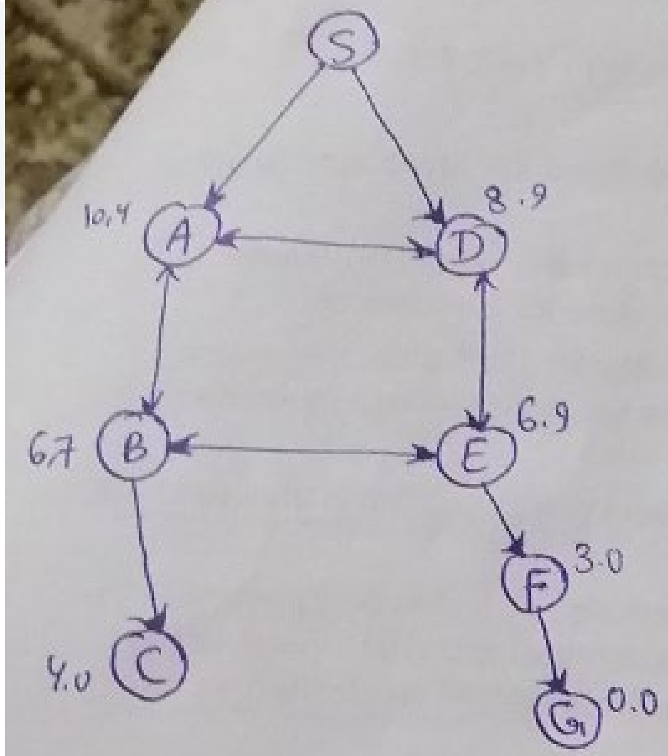
```
beamSearch(problemSet,ruleSet,
memorySize)
openMemory = new memory of size
memorySize
nodeList = problemSet.listOfNodes
node = root or initial search node
Add node to openMemory;
while (node is not a goal node)
Delete node from openMemory;
Expand node and obtain its children,
evaluate those children;
If a child node is pruned according to a
rule in ruleSet, delete it;
Place remaining, non-pruned children into
openMemory;
If memory is full and has no room for new
nodes, remove the worst
node, determined by ruleSet, in
openMemory;
node = the least costly node in
openMemory;
```

يحتفظ البحث الشعاعي بأفضل العقد (التي لها أقل قيمة للتابع الحدسي) في المجموعة open، عندما يكون حجم الشعاع 1 فتصبح الخوارزمية خوارزمية تسلق الهضبة، وعندما يكون حجم الشعاع غير منته يصبح البحث الأفضل أولاً Best First Search.

لنفهم الخوارزمية سنطبق على المثال انطلاقاً من العقدة S حتى العقدة G الآتي بحجم شعاع هو2:

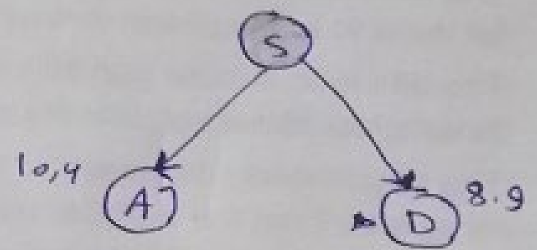


معامل التفرع = 2

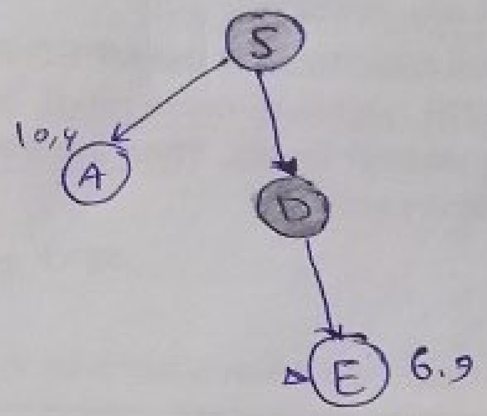


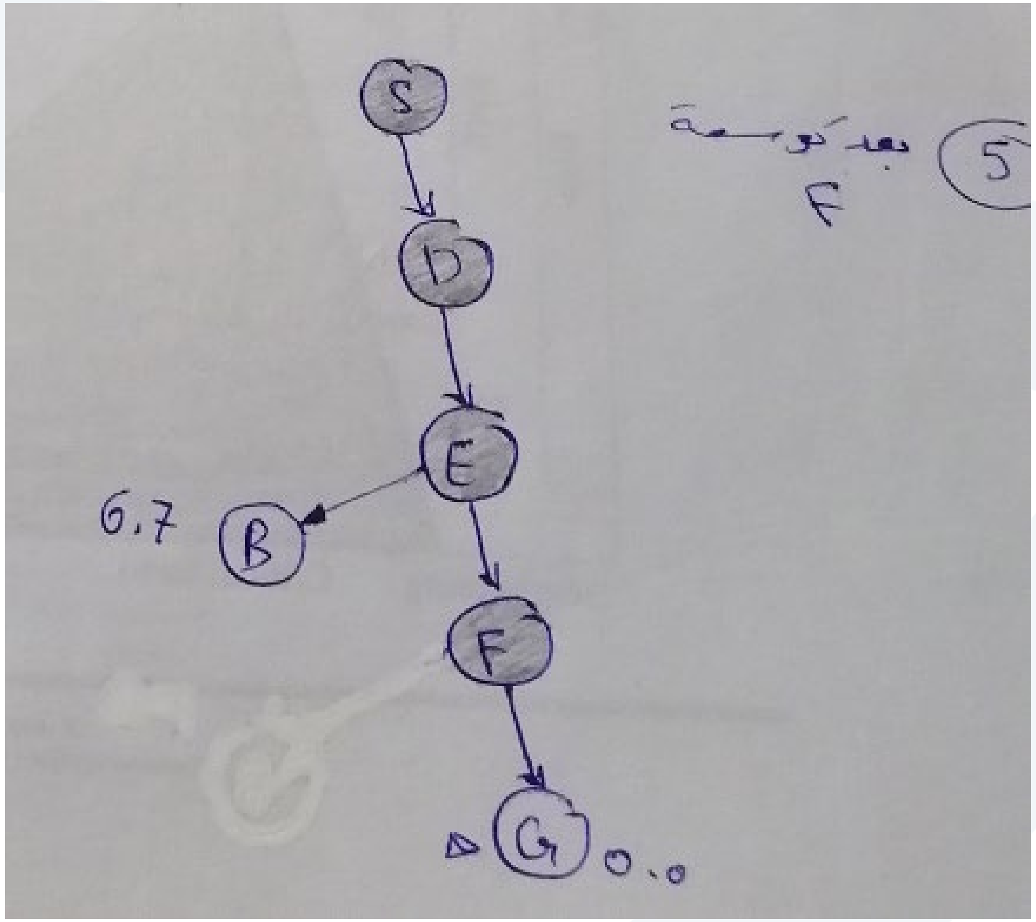
1 الحالة الابتدائية S

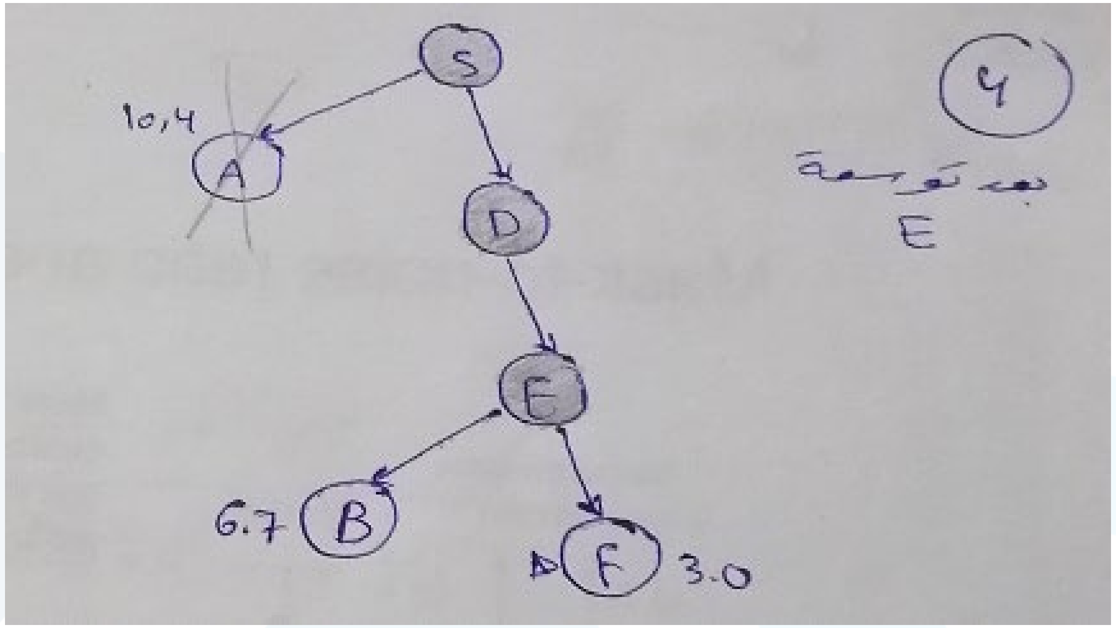
2 بعد توسعة S



3 بعد توسعة D

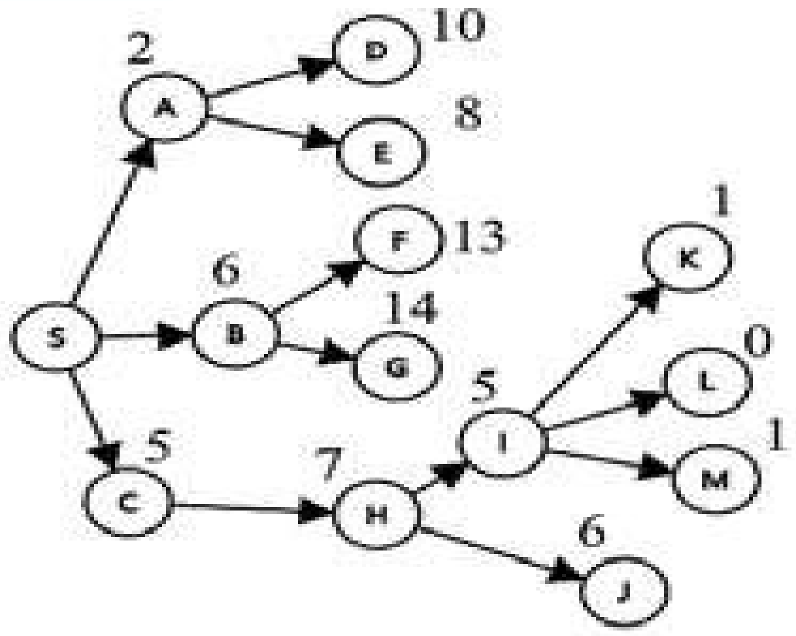



5
بعد توسعة
F
S
D
E
6.7
B
F
G
0.0


٤
بعد توسعة
E
S
10,4
A
D
E
6.7
B
F
3.0

Open	Closed
S	-
$D_{8.9}$ $A_{10.4}$	S
$E_{6.9}$ $A_{10.4}$	S $D_{8.9}$
$F_{3.0}$ $B_{6.7}$ $A_{10.4}$	S $D_{8.9}$ $E_{6.9}$
$G_{0.0}$ $B_{6.7}$	S $D_{8.9}$ $E_{6.9}$ $F_{3.0}$
Stop	S $D_{8.9}$ $E_{6.9}$ $F_{3.0}$ $G_{0.0}$

كما نلاحظ هي نفسها طريقة Best-First Search ولكن عندما يزيد عدد عناصر المجموعة Open عن حد الشعاع المحدد، نحذف العناصر الأكثر تكلفة حتى يبقى عنصرين.



تمرين 2:

طبق البحث الشعاعي على البيان الآتي علماً أن عقدة البداية هي S وعقدة النهاية هي L:

حجم الشعاع = 2

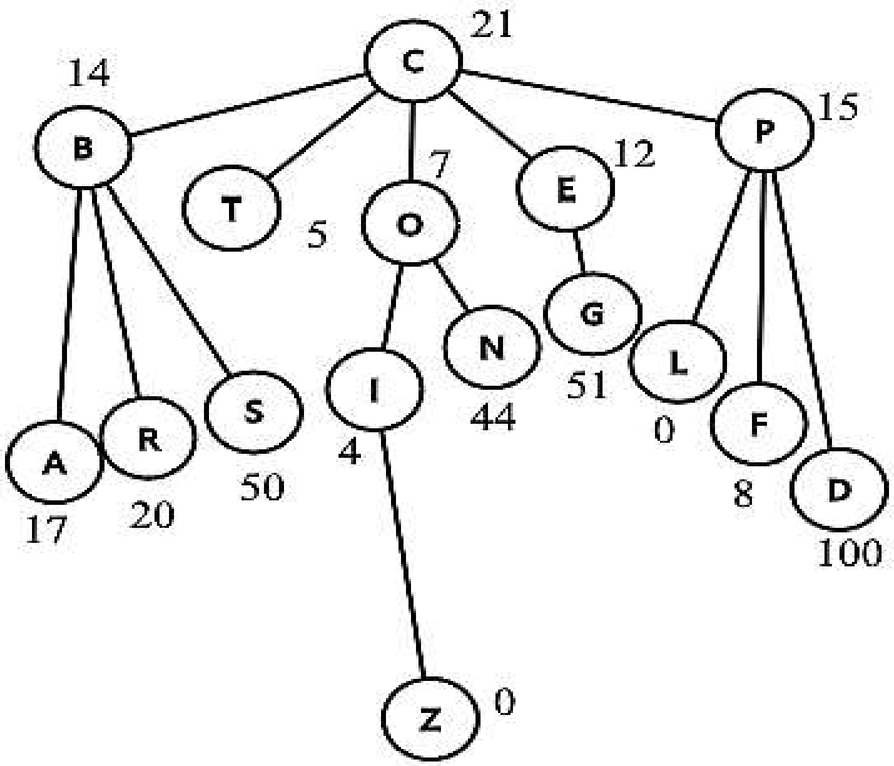
Open	Closed
S	-
A_2 C_5	S
C_5 E_8	S A
H_7 E_8	S A C
I_5 J_6	S A C H
L_0 K_1	S A C H I
Stop	S A C H I L

حجم الشعاع = 3

Open	Closed
S	-
A_2 C_5 B_6	S
C_5 B_6 E_8	S A
B_6 H_7 E_8	S A C
H_7 E_8 F_{13}	S A C B
I_5 J_6 E_8	S A C B H
L_0 K_1 M_1	S A C B H I
Stop	S A C B H I L

تمرين 3:

طبق خوارزمية البحث الشعاعي على البيان التالي علماً أن Cعقدة البداية و L,Z عقد النهاية وحجم الشعاع هو 2:



Open	Closed
C	-
T_5 O_7	C
O_7	C T
I_4 N_{44}	C T O
Z_0 N_{44}	C T O I
Stop	C T O I Z