

99521271

HW2-AI (بصن تئوری)

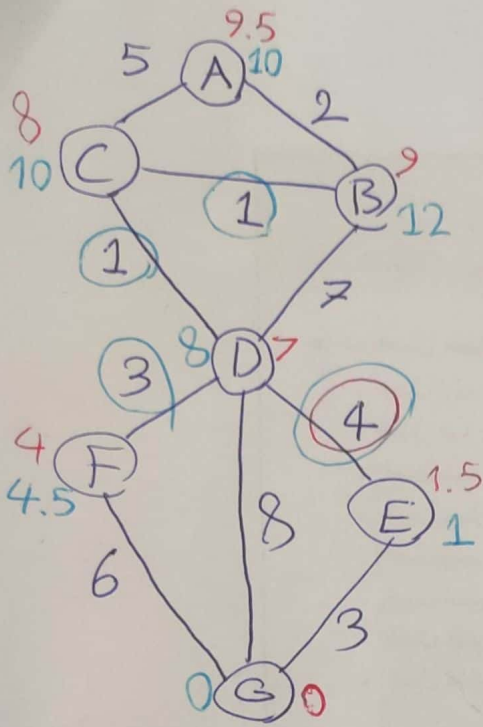
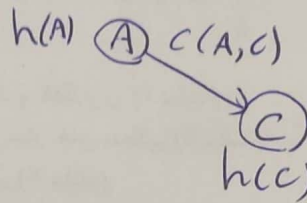
فرزان رحمانی

سوال 1

تعریف

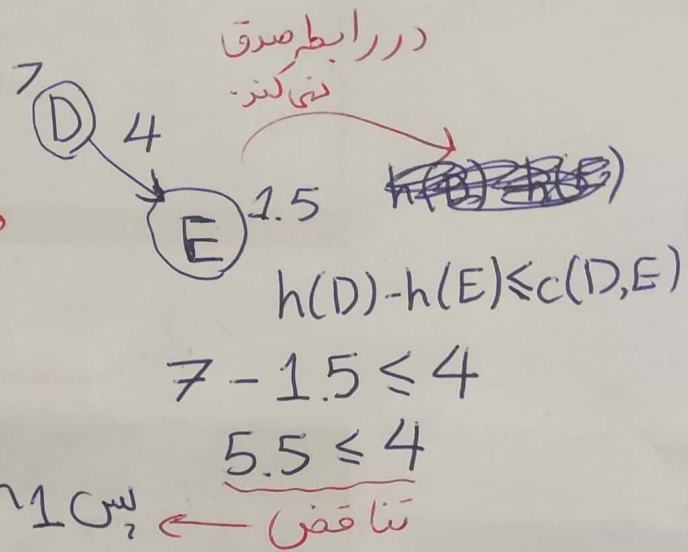
Consistency:  
heuristic arc cost  $\leq$  actual cost for each arc

$$h(A) - h(C) \leq c(A, C)$$



$h_1 \rightarrow$  قمر

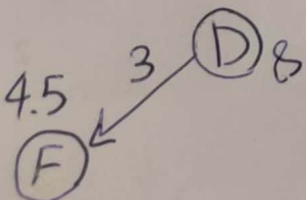
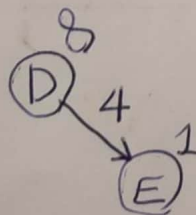
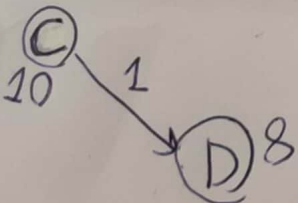
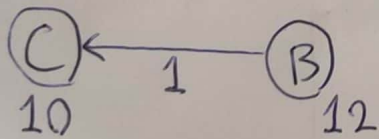
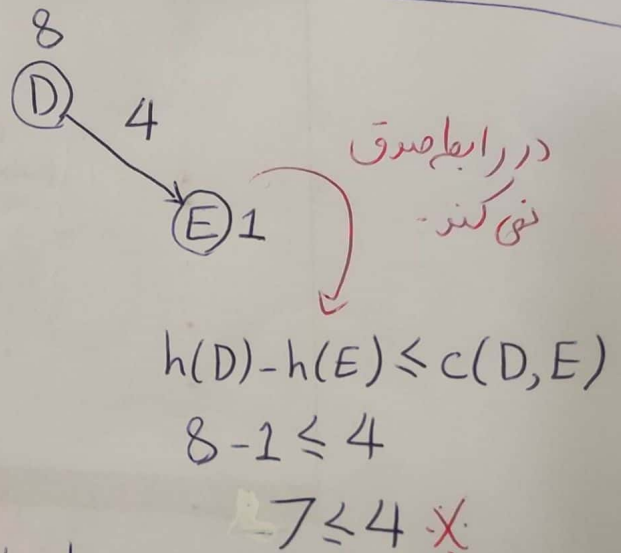
$h_1$   
consistent  
نیست

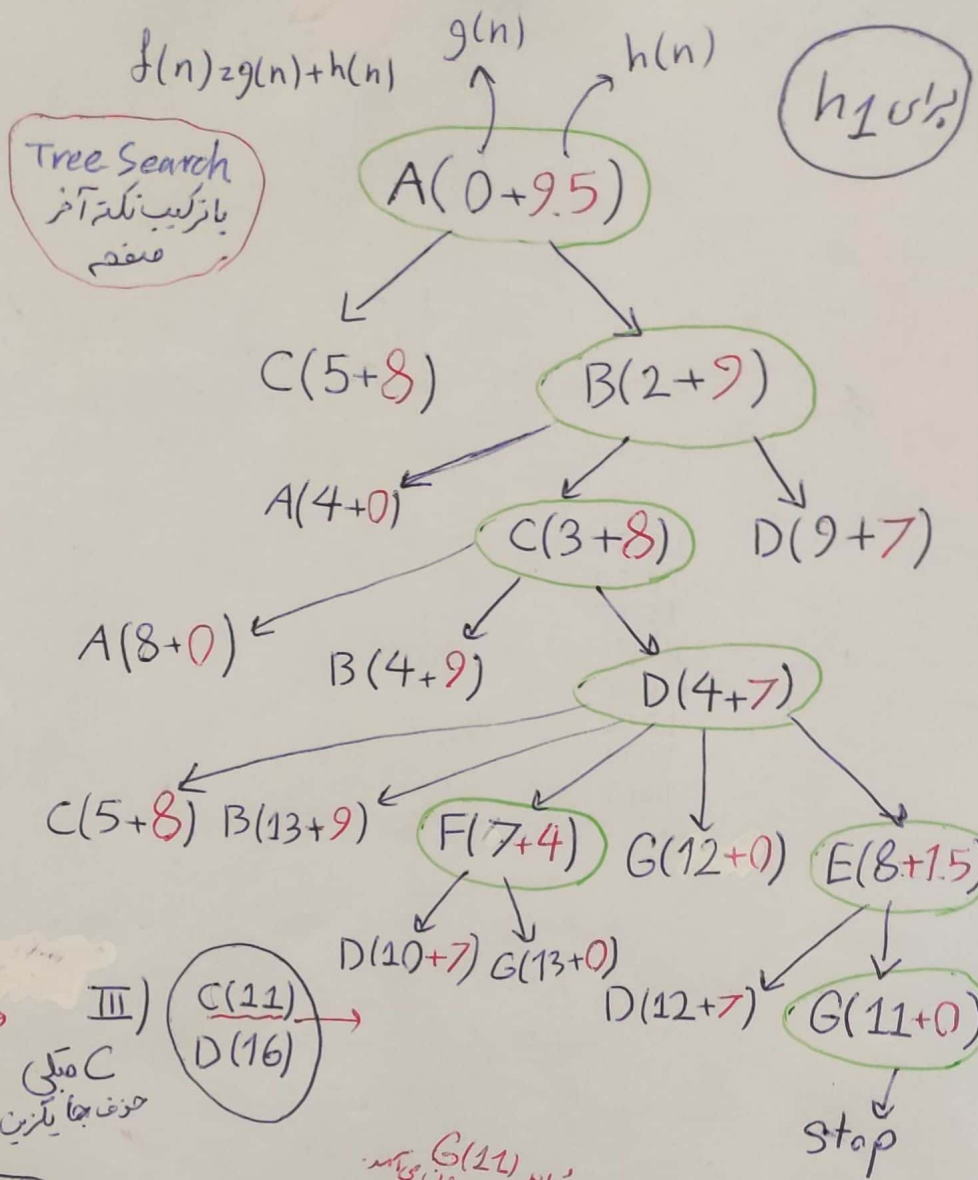
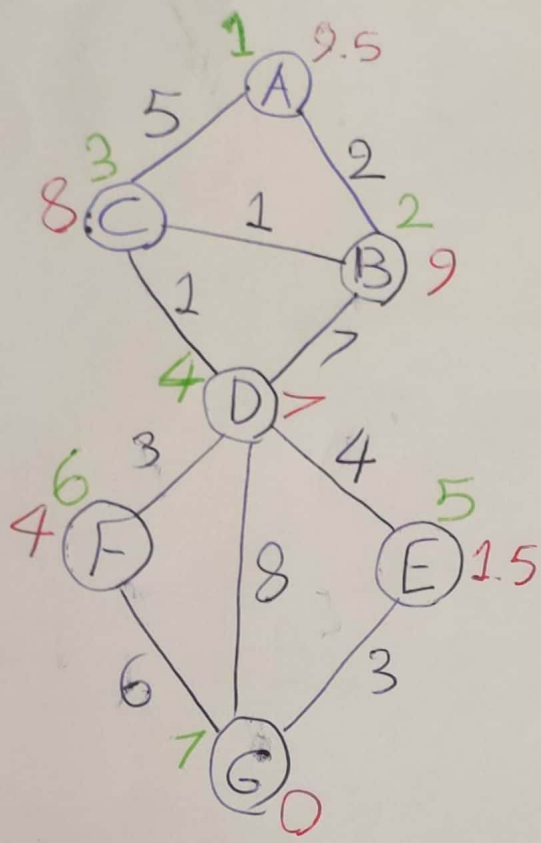


$h_2 \rightarrow$  آبی کم زنده

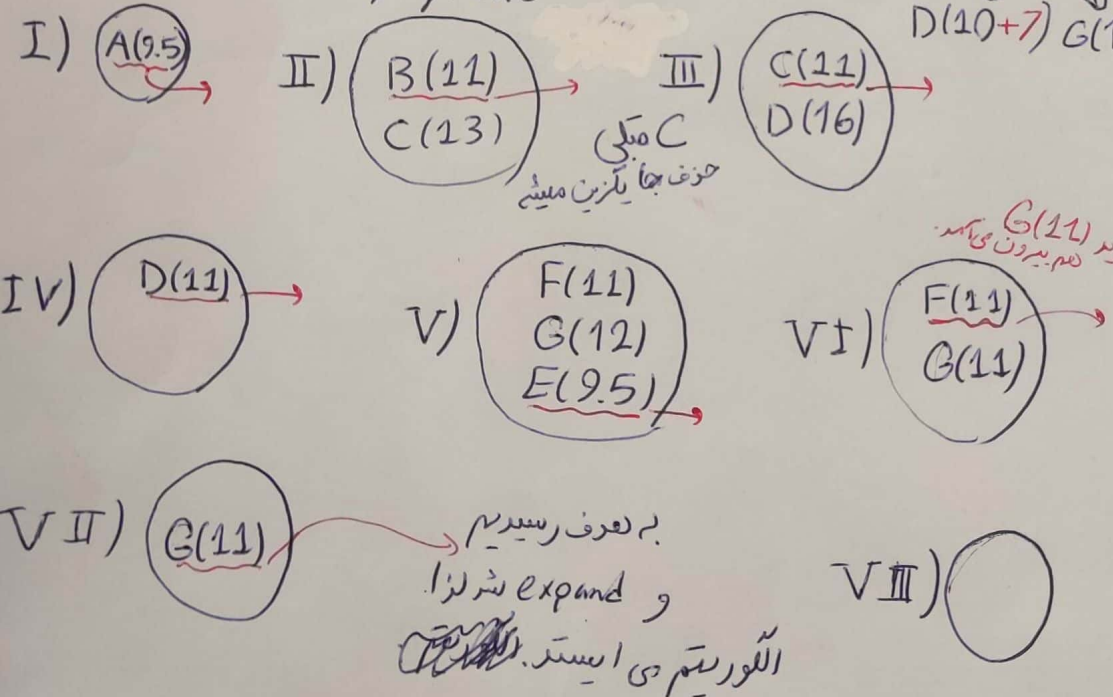
به علت این گمان ها

$h_2$   
Consistent  
نیست





Frontier: priority queue



از یک آرایه برای نگهداری  $f(n)$  هر node استفاده می کنیم که می تواند update شود.

ترتیب گره های  $\rightarrow$  A, B, C, D, E, F, G  
 الگوریتم می ایستد  
 expand شده

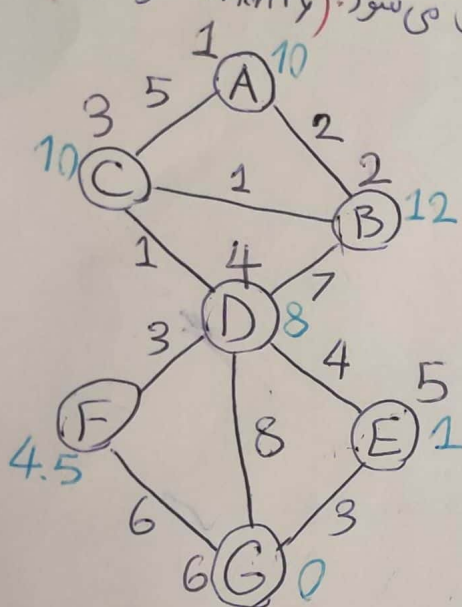
مسیر پیدا شده:  $A \xrightarrow{2} B \xrightarrow{1} C \xrightarrow{1} D \xrightarrow{4} E \xrightarrow{3} G$

priority queue (frontier) بدون گره عمل می کند که اگر node جدیدی را دیدیم به شرطی وارد priority queue می شود که نسبت به قبل  $f(n)$  کمتری داشته باشد در غیر این صورت وارد نمی شود همچنین اگر در frontier مقدار ترتیبی باشد مقدار حد درجا کمتر از آن می شود و قبلی  $f(n)$  معتبر نیست.



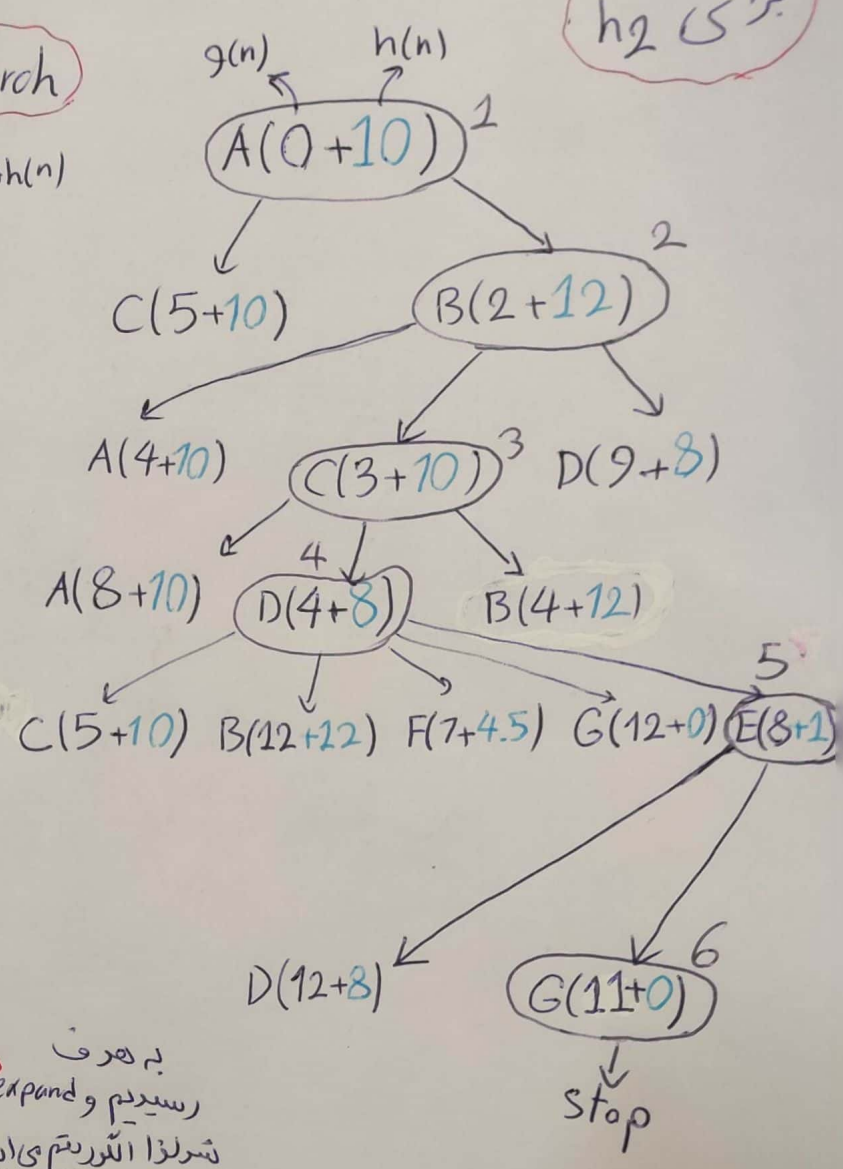
از یک آرایه برای نگهداری  $f(n)$  هر node استفاده می کنیم که ابتدا  $\infty$  است و update می شود

frontier (priority queue)  $f(n)$  کمتری داشته باشد در غیر این صورت وارد نمی شود و همچنین اگر در frontier مقدار بزرگتری باشد مقدار جدید و کوچکتر  $f(n)$  جایگزین می شود. (Change Priority)



**A\* tree search**  
 $f(n) = g(n) + h(n)$

برای  $h_2$



Frontier: priority queue

- I) A(10)
- II) B(14)  
C(15)
- III) C(13)  
D(17)
- IV) D(12)
- V) F(11.5)  
G(12)  
E(9)
- VI) G(11)  
F(11.5)
- VII) F(11.5)

به هدف رسیدیم و expand نشد. بنابراین الگوریتم ایستاد.

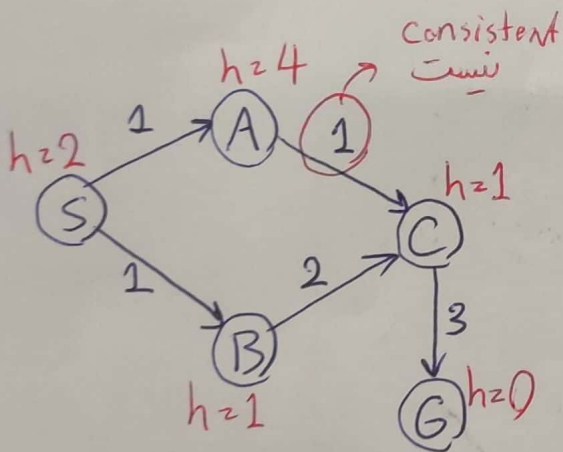
ترتیب گره های expand شده: A, B, C, D, E, G

مسیر پیراشده:  $A \xrightarrow{2} B \xrightarrow{1} C \xrightarrow{1} D \xrightarrow{4} E \xrightarrow{3} G$

الف) نادرست - بازی سودو کو یک بازی deterministic است چرا که با احتمالات سروکار ندارد و حالت بعدی محیط کاملاً به حالت فعلی و عمل ما بستگی دارد.  
 $\text{action}$   $\text{current state}$   $\text{next state}$

ب) درست - چون که پیچیدگی فضایی DFS چنجه ای است ولی پیچیدگی BFS نمایی است.  
 $BFS: O(b^s) \rightarrow$  نمایی  
 $DFS: O(bm) \rightarrow$  چنجه ای

$s = \text{smallest depth of solution}$   
 $m = \text{max depth of search tree } (s \leq m)$   
 $b = \text{branching factor}$



پ) نادرست است زیرا مثال نقض زیر را دارد.

$h$  admissible هست ولی consistent نیست

تعریف  $\rightarrow h(A) - h(C) \leq c(A, C)$

$$4 - 1 \leq 1$$

$$3 \leq 1$$

تناقض بالتعریف

Consistency  $\rightarrow$  admissibility

ت) درست است.

چون که این ویژگی قوی تر و سخت تر از consistency

Consistency Implies admissibility

ویژگی admissibility است و در اسلایدها آمده است

طبق صفحه 11 اسلاید AI-04



سوال ۳ بهی توان با الوریتم  $A^*$  که یک نوع informed search است

این پازل را حل کرد. به این شکل که هر یک از state هایی که اعداد در خانه ها قرار گرفته اند را یک node گراف در نظری بگیریم و هر یک از action های left, right, down, up برای خانه خالی را edge گراف در نظری بگیریم. حال start state بازی را به عنوان root در tree search در نظری بگیریم. frontier الوریتم  $A^*$  یک صف اولویت دار بر اساس  $f(n)$  است. cost هریال را یک در نظری بگیریم.  $f(n) = g(n) + h(n)$  تعریف می کنیم.

مجموع cost های یال ها از start state تا node فعلی

$g + h$   
Total Manhattan distance from current state to the goal

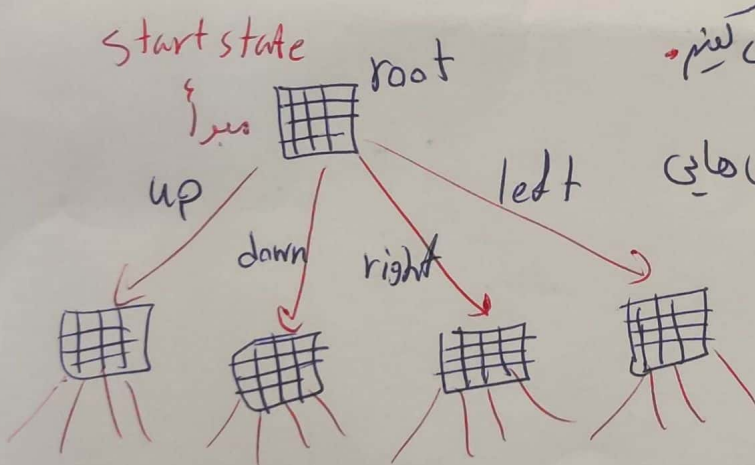
حال با تعریف  $f(n)$  الوریتم  $A^*$  را شروع می کنیم تا به goal state برسیم و سپس از start state

یال های این مسیر action sequences را برای بردن بازی به ما می دهد. زمانی که به


رأس goal state رسیدیم الوریتم را stop می کنیم.

\* برای heuristic می شود از تعداد کاشی هایی که سر جای خود نیستند هم استفاده کرد ولی manhattan distance مربع تراست.


$A^*$  کوتاه ترین مسیر (با کمترین تعداد حرکت) را پیدا می کند.



nodes ... ,  ,  ,  , ... حالت های state Space: خانه ها:

initial State:  → root

Actions: up, down, left, right → edges

Goal test (Goal state):  خانه ها مرتب شده اند

$A^*$  (informed search) چون که  $h$ ، admissible هست و  $g$  هم  $h$  را برآورد می‌دهد، پس  $A^*$  هم Optimal و هم  $h$  را برآورد می‌دهد، پس  $A^*$  هم Complete هم می‌باشد.

Ques 5  $\mathbb{Z}$  complete  $\mathbb{Z}$

Complexity این الوریتم وابسته به heuristic می باشد.

در بدترین حالت تعداد node های توسعه داده شده جهانی است ولی در عمل  
یا یک heuristic خوب بسیار سریع و خوب عمل می کند.

$A^{\star}$ : Time:  $O(b^d) = O(b^{c^{\star}/\epsilon}) = O(4^d) = O(4^{c^{\star}/\epsilon})$   
Space:  $O(b^d) = O(b^{c^{\star}/\epsilon}) = O(4^d) = O(4^{c^{\star}/\epsilon})$

worst case In Theory

- $b = \text{branching factor} = 4$
- $d = \text{depth of shortest path}$
- $C^* = \text{Cost of optimal solution}$
- $E = \text{minimum cost between 2 nodes}$

$\frac{C^*}{C} \rightarrow$  effective depth