AI. HW 1: Search and games

1. Informed search and A*.

a) States: $\{(a,b) : a,b \in \{1,2,3,4,5\}\}$. $s0 = (1,1)$. $sf = (5,5)$

Operators: North, South, East, West. No preconditions.

Postconditions:
North $(a,b) \rightarrow (a, b+1)$ if $b<5$, else (a,b)
South $(a,b) \rightarrow (a, b-1)$ if $b>1$, else (a,b)
East $(a,b) \rightarrow (a+1, b)$ if $a<5$, else (a,b)
West $(a,b) \rightarrow (a-1, b)$ if $a>1$, else (a,b)

Cost associated: Walls = $\{(1,2),(2,2),(3,2),(5,2),(2,4),(3,4),(4,4),(5,4)\}$

If prev_state $\notin$ Walls and next_state $\in$ Walls: cost = 2

Else: cost = 1.

b) i) min branching factor: 2 (the corners). max br. factor: 4.

ii) Avoid cycles: use a graph alg. (remove repeated nodes).

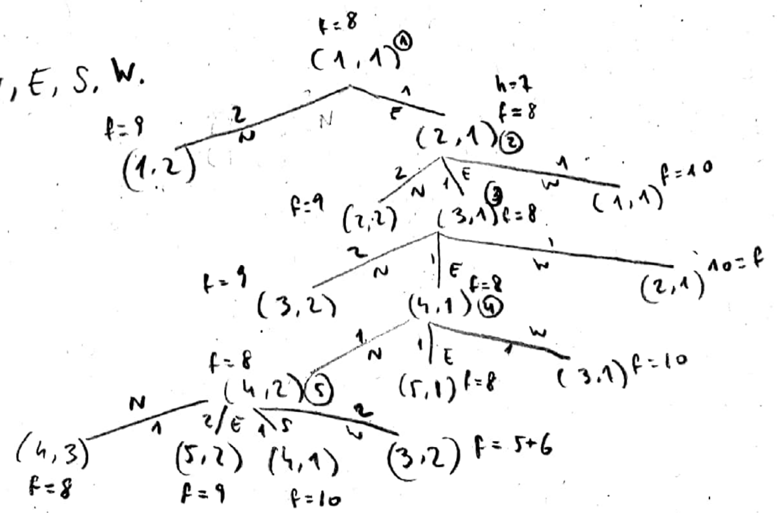
c) Admissible heuristic: $h(a,b) = |5-a| + |5-b| = 10-(a+b)$

If the cost was 1 for every jump, $h(a,b)$ would be exactly the min cost to reach (5,5). As the cost is higher or equal to 1, $h(a,b)$ is lower or equal to $g^*(a,b)$. Therefore, it's admissible.

d) Yes. Because h is consistent: If n' is successor of n, then $h(n') = h(n) \pm 1 \geq h(n)-1$ and $cost(n \to n') \in \{1,2\} \Rightarrow cost(n \to n') \geq 1 \Rightarrow h(n') + cost(n \to n') \geq h(n) - 1 + 1 = h(n)$

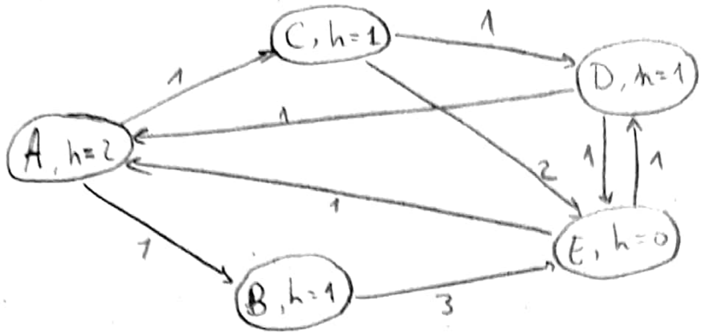
e) Yes.

f) order: N, E, S, W.



2. Search with A*

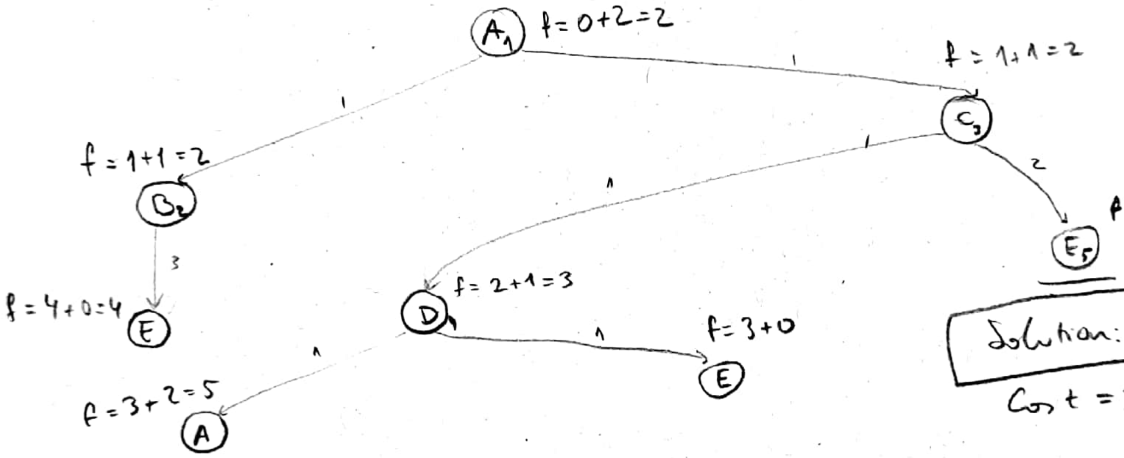
i)



<u>h is monotonic:</u>

Node:	Successors	
A	B, C	$\rightarrow h(A) = 2 \le \begin{cases} 1 + h(C) \\ 1 + h(B) \end{cases}$
B	E	$\rightarrow h(B) = 1 < cost(B,E) + h(E) = 3$
C	D, E	$\rightarrow h(C) = 1 < \begin{cases} 1 + h(D) \\ 2 + h(E) \end{cases}$
D	A, E	$\rightarrow h(D) = 1 \le \begin{cases} 1 + h(A) \\ 1 + h(E) \end{cases}$
E	A, D	$\rightarrow h(E) = 0 < \begin{cases} 1 + h(D) \\ 1 + h(A) \end{cases}$ $\square$

ii) Beside each node: $f = g + h$ (notation). Subindex: order of expanded nodes.



$\boxed{\text{Solution: } A \rightarrow C \rightarrow E}$

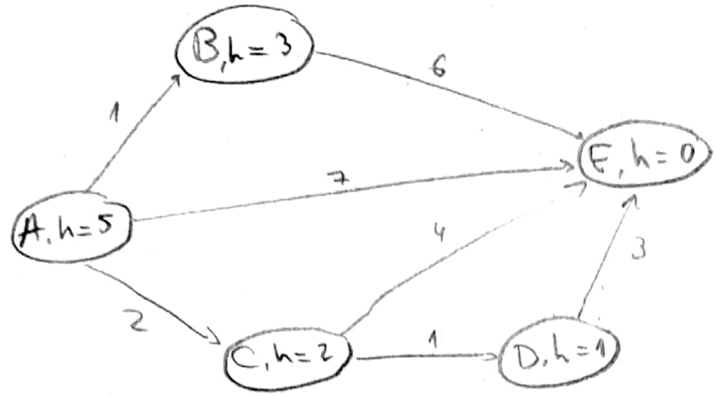
Cost = 3.

iii) This search has found an optimal path.

The heuristic defined is consistent (i), so the graph search guarantees to find an optimal path.

3. A* Search

i)

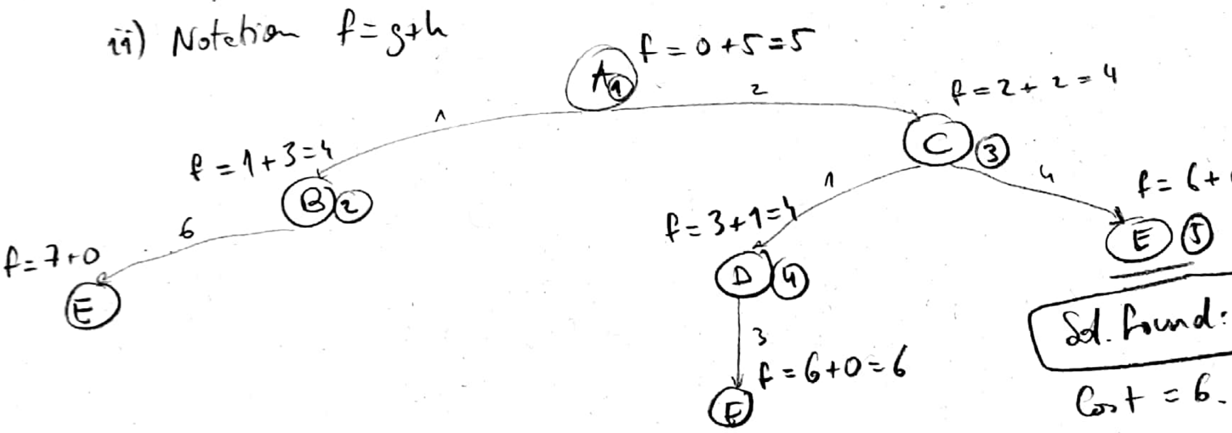


h is not monotonic: $h(A) = 5 > cost(A,B) + h(B) = 1 + 3 = 4$.

h is admissible: Let $M(node) = $ min. cost path (node, E)

$$\begin{cases} h(goal) = h(E) = 0 \\ h(A) = 5 \leq M(A) = 6 \\ h(B) = 3 \leq M(B) = 6 \\ h(C) = 2 \leq M(C) = 4 \\ h(D) = 1 \leq M(D) = 3 \end{cases}$$

ii) Notation $f = g + h$



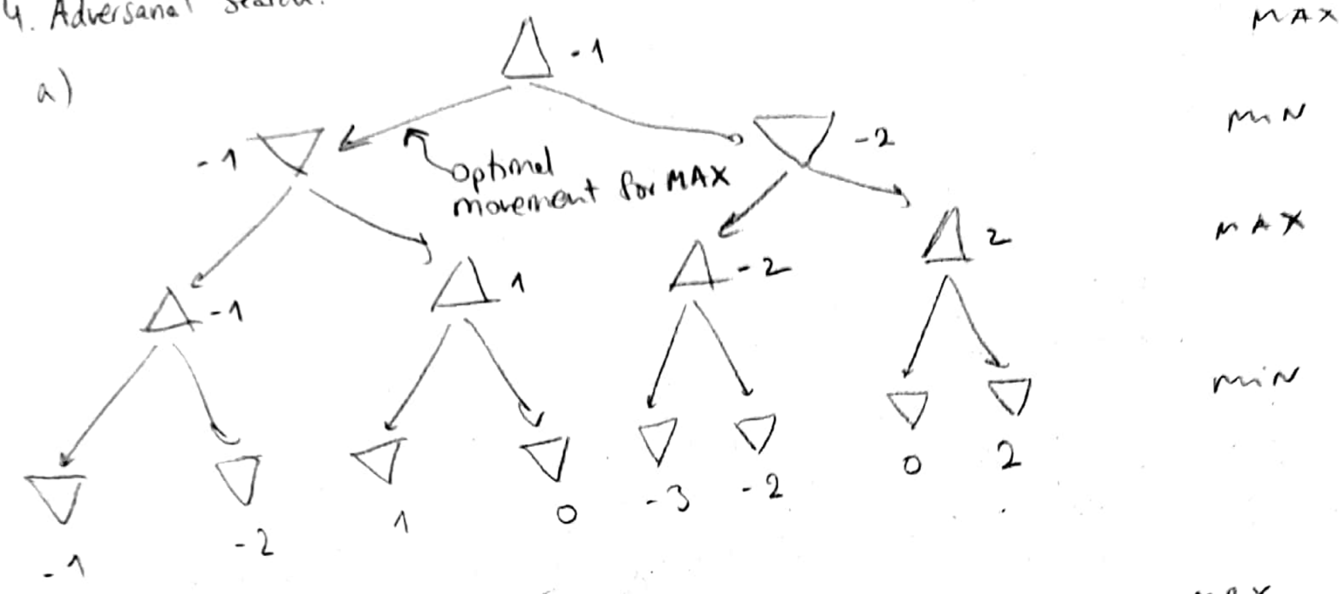
Sol. found: A→C→E

Cost = 6.

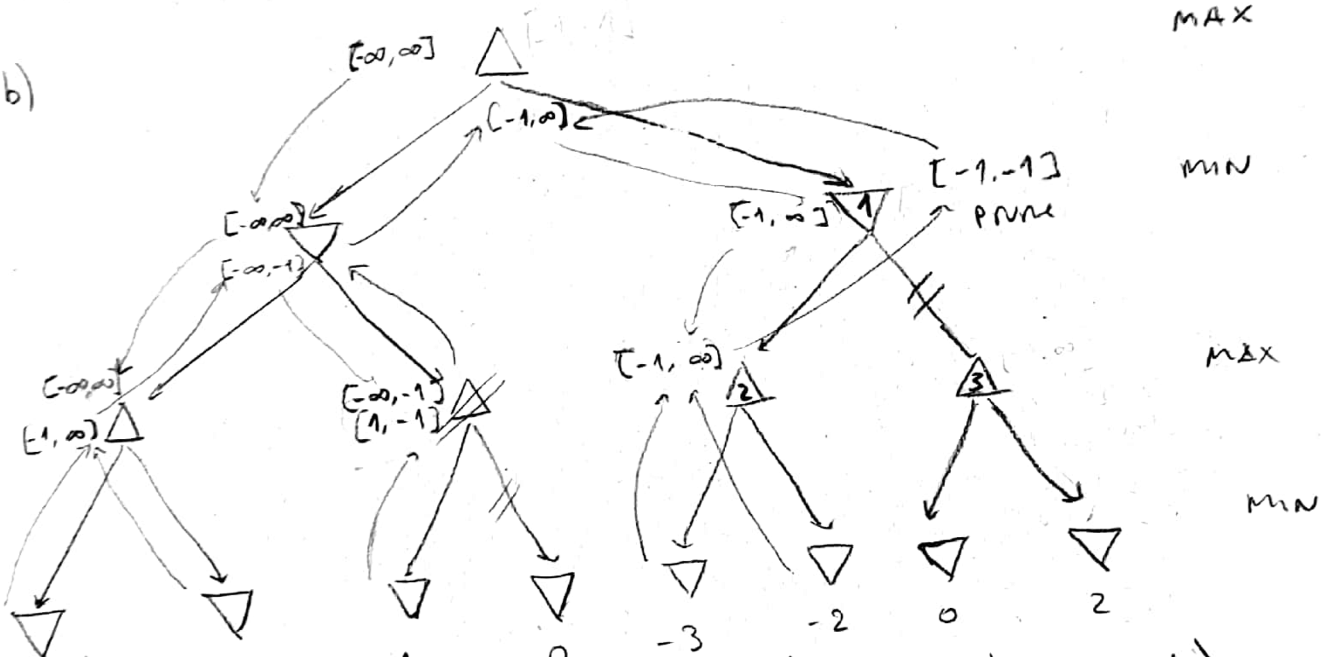
iii) The algorithm has found an optimal path. A* does not guarantee, with elimination of repeated states, to find an optimal path, but a non-monotonic heuristic,

4. Adversarial Search.

a)



b)



(Right has been pruned).

Root: -1 ; go left

c) A = B = 1, C = D = 0

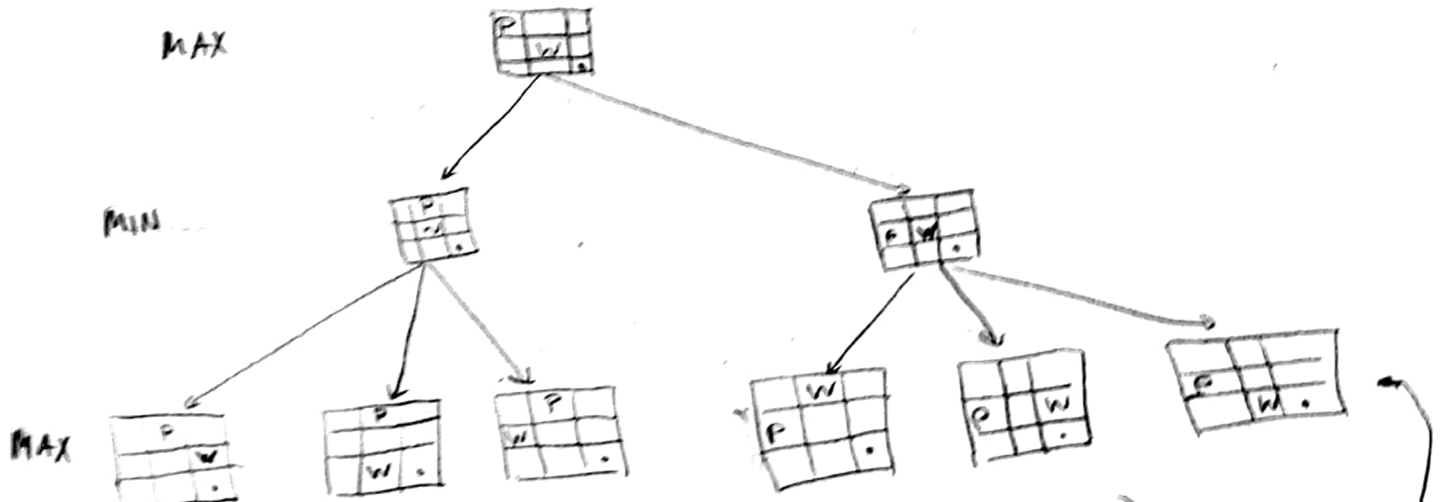
This way, node 2 would have: [1, ∞) ⟶ node 1 [-1, 1]

⟶ node 3 [-1, 1] ⟶ node 3 [-∞, 1] ⟶ node 1 [-1, 0)

i.e. no pruning

5 Adversarial search

a,b) P ≡ ☺, W = [box], • ≡ • Order: ↑, →, ↓, ←

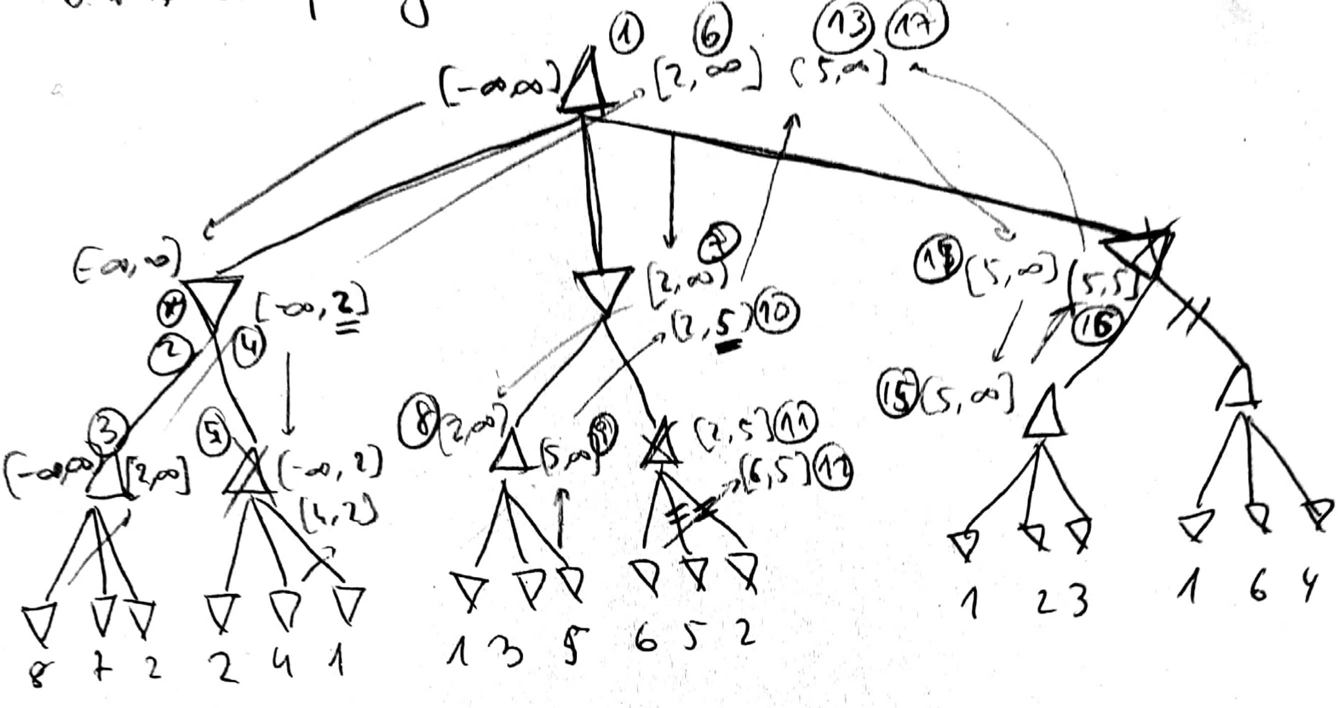


c) Pacman has eaten 0 dots in every state shown in a), so every node has a value of 0.

d) The value would be 1. Pacman here can always choose a move to eat the dot, whatever the W does.

e,f)

6. Alfa - beta pruning



a) Minimax value at root: 5

b) Optimal play: left ⊛

c) Left to right

d) MAX: Δ, MIN: ∇

e) Doesn't look like a 0-sum game

f) Minimax does not guarantee the optimal strategy, because it relies on the assumption that MIN plays optimally.

HW6