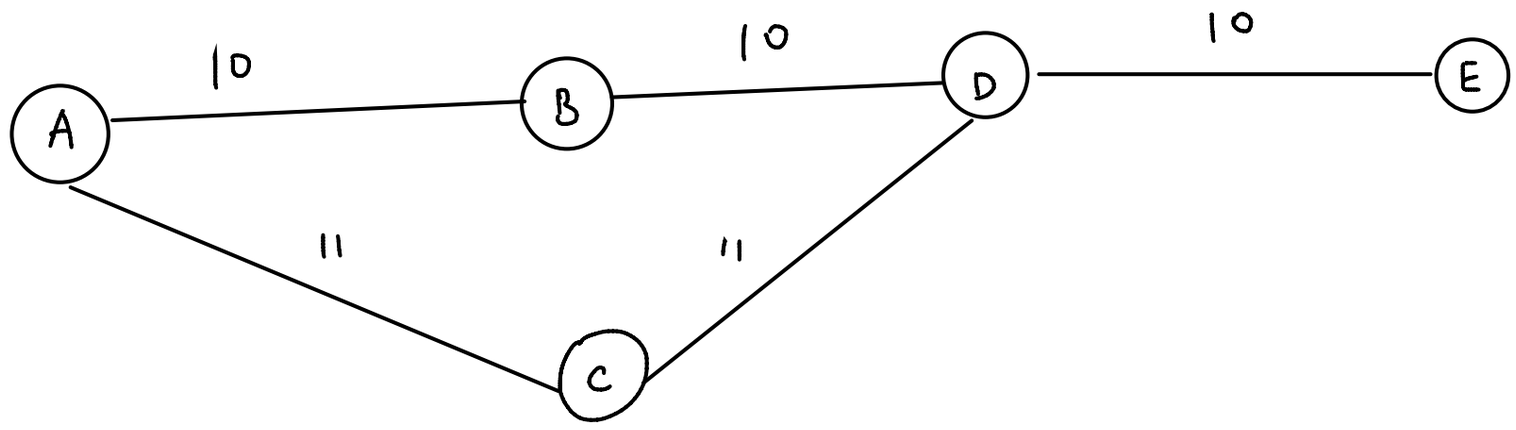
Uniform Cost Search

Graph - Search

* Frontier = { Start State / Vertex }
* do
  * If Frontier = Empty ⇒ Search failed (Priority Queue with LEAST Path Cost as priority)
  * Pick a node ∈ Frontier, Frontier = Frontier - Node (Selection)
  * If node = Goal the stop (Goal Test) (trace route back to start)
  * Explored = Explore + node.state
  * Frontier = Frontier + children(node) (Expansion)
    - children(node) ∩ Explored
    - children(node) ∩ Frontier

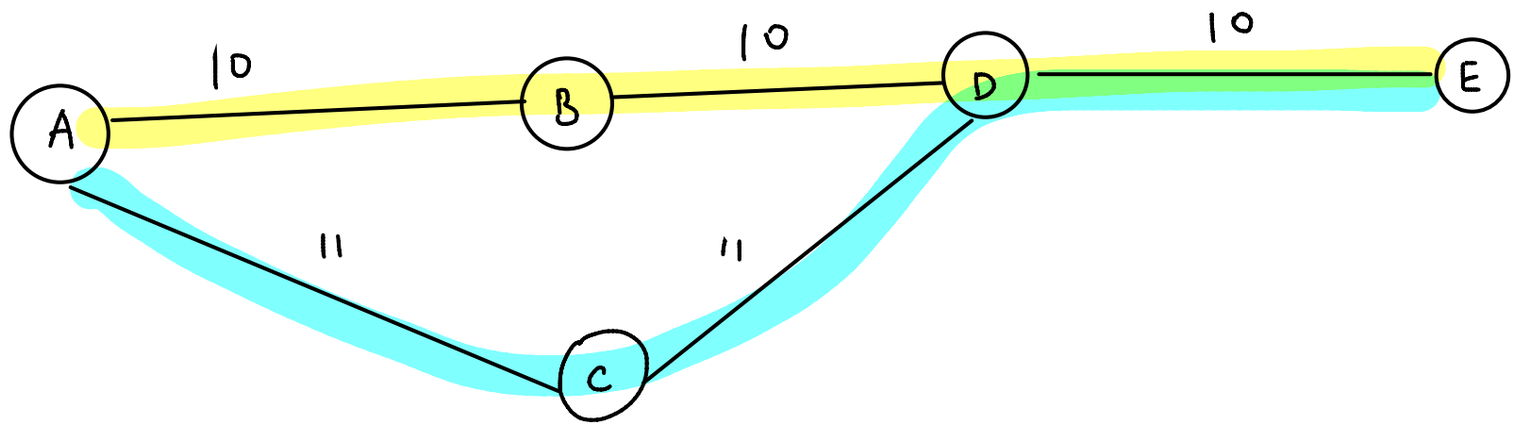
} Absent in Tree Search

If node is already present in the frontier with high path cost than current, then replace it in the frontier with current node



Uniform cost Search (Tree - Search)
loops not avoided

	Frontier node	total cost to reach node	Top priority
Step 1	$n_1 = A$	0	n_1
step 2	$n_2 = B$	10	
	$n_3 = C$	11	
step 3	$n_3 = C$	11	n_2
	$n_4 = D$	20	
	$n_5 = A$	20	



Uniform cost Search (Tree - Search)
loops not avoided

	node	total cost to reach node	Top priority
Step 4	$n_4 = D$	20	n_3
	$n_5 = A$	20	
	$n_6 = D$	22	
	$n_7 = A$	22	

Top pair being

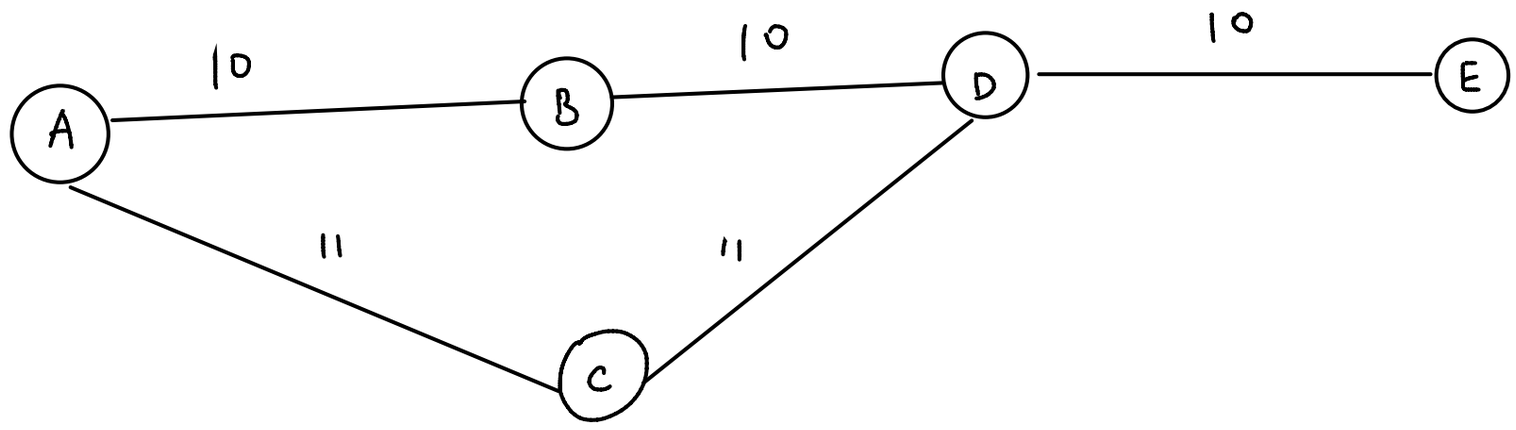
	node		total cost to reach node	
Step 5	n_5	= A	20	n_4
	n_6	= D	22	
	n_7	= A	22	
	n_8	= E	30	
	n_9	= B	30	
Step 6	n_6	= D	22	n_5
	n_7	= A	22	
	n_8	= E	30	
	n_9	= B	30	
	n_{10}	= B	30	
	n_{11}	= C	31	

	node		total cost to reach node	Top priority
Step 7	n_7	= A	22	n_6
	n_8	= E	30	
	n_9	= B	30	
	n_{10}	= B	30	
	n_{11}	= C	31	
	n_{12}	= B	32	
	n_{13}	= E	32	

	node		total cost to reach node	Top priority
Step 8	n_8	= E	30	n_7
	n_9	= B	30	
	n_{10}	= B	30	
	n_{11}	= C	31	
	n_{12}	= B	32	
	n_{13}	= E	32	
	n_{14}	= B	32	
	n_{15}	= C	33	

node	total cost to reach node	Top priority
$n_9 = B$	30	n_8
Step 9		
$n_{10} = B$	30	
$n_{11} = C$	31	
$n_{12} = B$	32	
$n_{13} = E$	32	
$n_{14} = B$	32	
$n_{15} = C$	33	

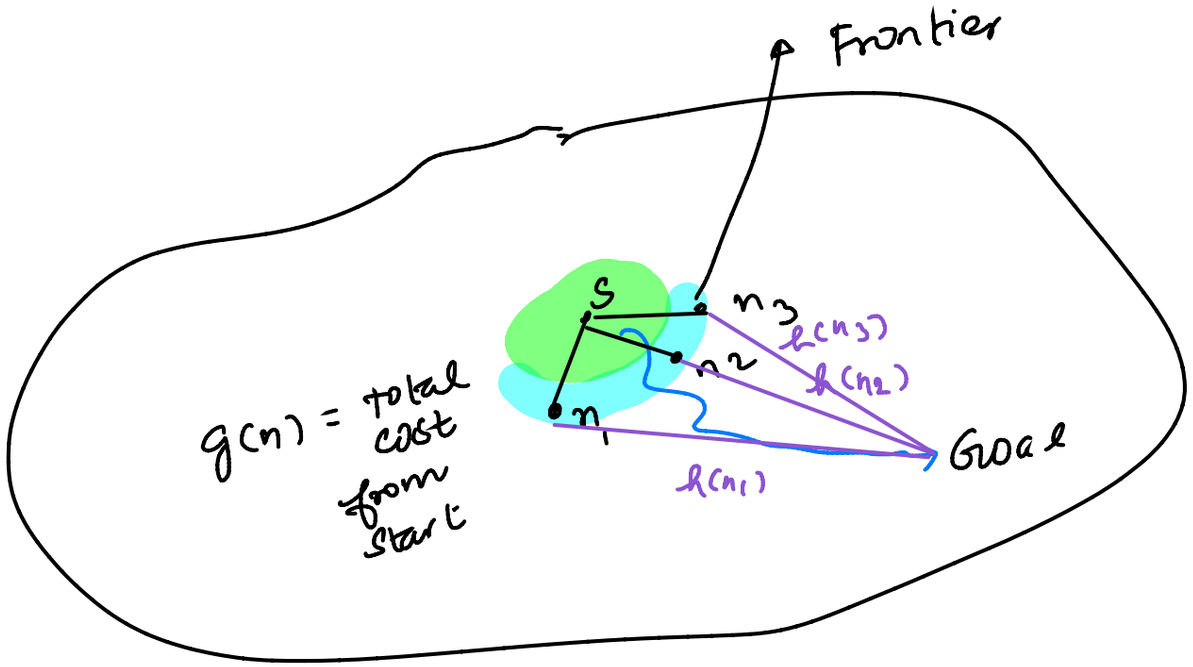
$n_8 = E$ is the Goal



Uniform cost Search (Graph - Search)
loops not avoided

	Frontier node	total cost to reach node	Top priority	Explored list
Step 1	$n_1 = A$	0	n_1	A
Step 2	$n_2 = B$	10	n_2	A, B
	$n_3 = C$	11		
Step 3	$n_3 = C$	11		
	$n_4 = D$	20		

	node	total cost to reach node	Top priority	Explored list
Step 4	n_4 = D	20	n_3	A, B, C
	avoided × n_5 = A			
	avoided × n_6 = D	21		
	n_5 = E	30	n_4	A, B, C, D
Step 5	avoided × n_6 = B		n_5	A, B, C, D
Step 6	n_5 = E is Goal state			



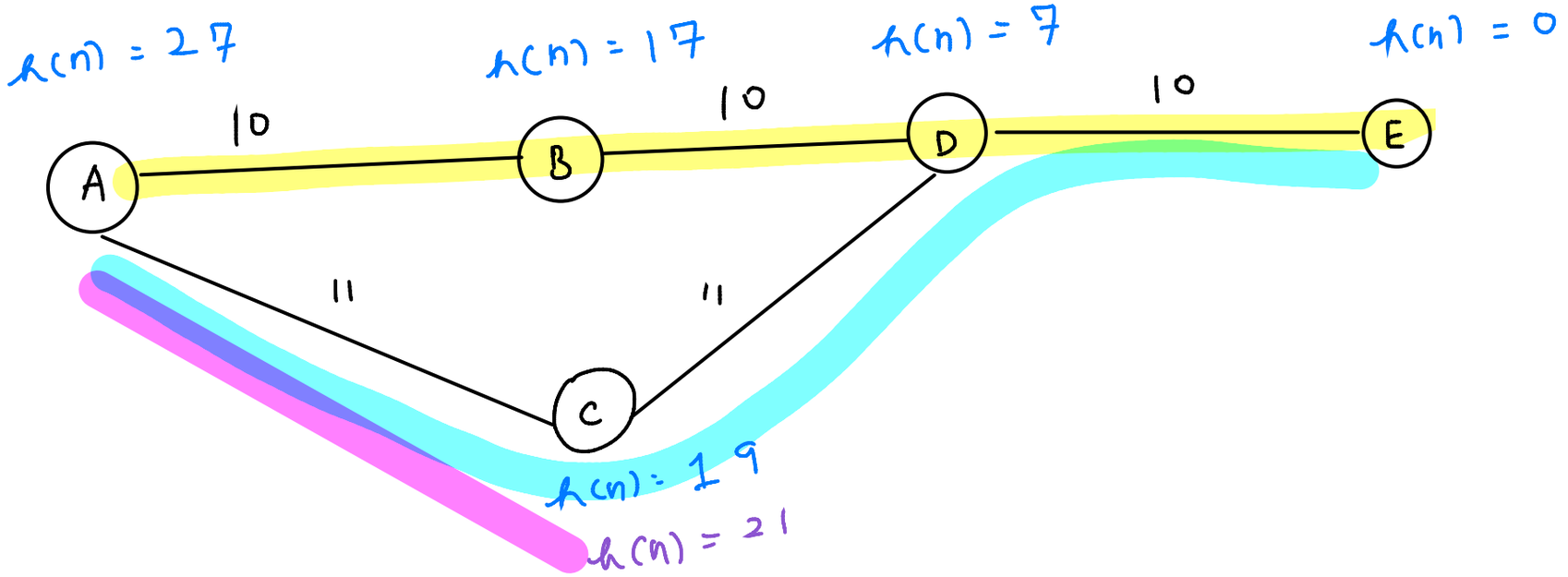
$g(n_1)$ vs $g(n_2)$ vs $g(n_3)$
$+$ $+$ $+$
$h(n_1)$ $h(n_2)$ $h(n_3)$

Expand the frontier on the most promising line

Uniform cost = $g(n)$ (total cost of the path till node)

Heuristic Search : $g(n) + \underbrace{h(n)}_{\text{approximate cost from node to Goal}}$

$\underbrace{\qquad\qquad}_{\text{Approximate total cost from Start to Goal}}$



Heuristic : Admissible

$h(n)$: actual cost from n to Goal

Tree - Based Search

	Frontier	$g(n)$ Total cost till n	$h(n)$ Heuristic cost	$f(n) = g(n) + h(n)$	Top priority
Step 1	$n_1 = A$	0	27	27	n_1
Step 2	$n_2 = B$	10	17	27	
	$n_3 = C$	11	19	30	

	Frontier	g(n) Total cost so far	h(n) Heuristic cost	f(n) = g(n) + h(n)	Top priority
Step 3	n_3 = C	11	19	30	n_2
	n_4 = D	20	7	27	
	n_5 = A	20	27	47	
Step 4	n_3 = C	11	19	30	n_4
	n_5 = A	20	27	47	
	n_6 = E	30	0	30	
	n_7 = B	30	17	47	
Step 5	n_5 = A	20	27	47	n_3
	n_6 = E	30	0	30	
	n_7 = B	30	17	47	
	n_8 = D	22	7	29	
	n_9 = A	22	27	49	

steps	Frontier	$g(n)$ Total cost till	$h(n)$ Heuristic cost	$f(n) = g(n) + h(n)$	Top priority
	n_5 : A	20	27	47	
	n_6 : E	30	0	30	n_8
	n_7 : B	30	17	47	
	n_9 : A	22	27	49	
	n_{10} : E	30	7	37	
	n_{11} : B	30	17	47	
	n_{12} : C	31	19	50	

step 6 n_6

Example of wrong ≡ inadmissible heuristic