Heuristic Search

Jonathan Irvin Gunawan
National University of Singapore

modul di NUS

CS3243 - Introduction to Artificial Intelligence

prerequisite

tau graph

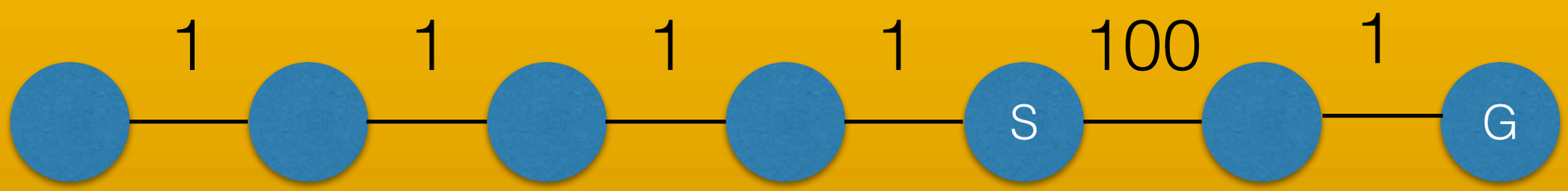
tau BFS+DFS

tau Dijkstra

motivation buat hari ini

ada gak yang (generally) lebih “cepet" dari Dijkstra?

kalau gak ada, ngapain ada materi hari ini?


1
1
1
1
100
1
S
G

ide : kita mo traverse dari yang paling “deket” dulu, most likely finish duluan

bikin f(n) untuk tiap node n.

f(n) = “estimate cost”

terus kita traverse nya based on increasing f(n)

terus kita traverse nya based on increasing f(n)

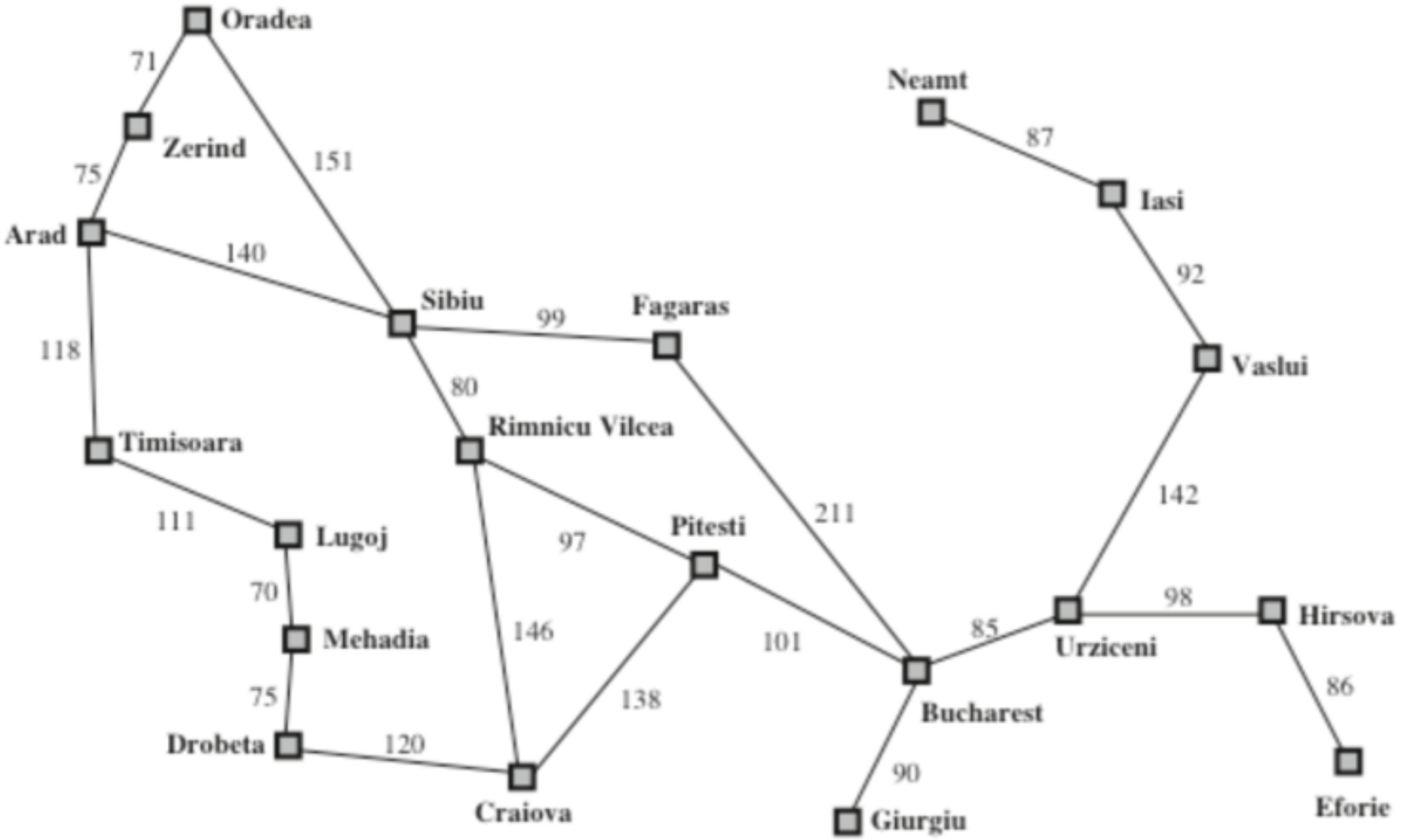
bisa pake pqueue

```
function UNIFORM-COST-SEARCH(problem) returns a solution, or failure

  node ← a node with STATE = problem.INITIAL-STATE, PATH-COST = 0
  frontier ← a priority queue ordered by PATH-COST, with node as the only element
  explored ← an empty set
  loop do
    if EMPTY?(frontier) then return failure
    node ← POP(frontier)   /* chooses the lowest-cost node in frontier */
    if problem.GOAL-TEST(node.STATE) then return SOLUTION(node)
    add node.STATE to explored
    for each action in problem.ACTIONS(node.STATE) do
      child ← CHILD-NODE(problem, node, action)
      if child.STATE is not in explored or frontier then
        frontier ← INSERT(child, frontier)
      else if child.STATE is in frontier with higher PATH-COST then
        replace that frontier node with child
```

kalo dijkstra, f(n) = jarak dari start ke n.

contoh : mau ke Bucharest



Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

kita bikin heurstic function

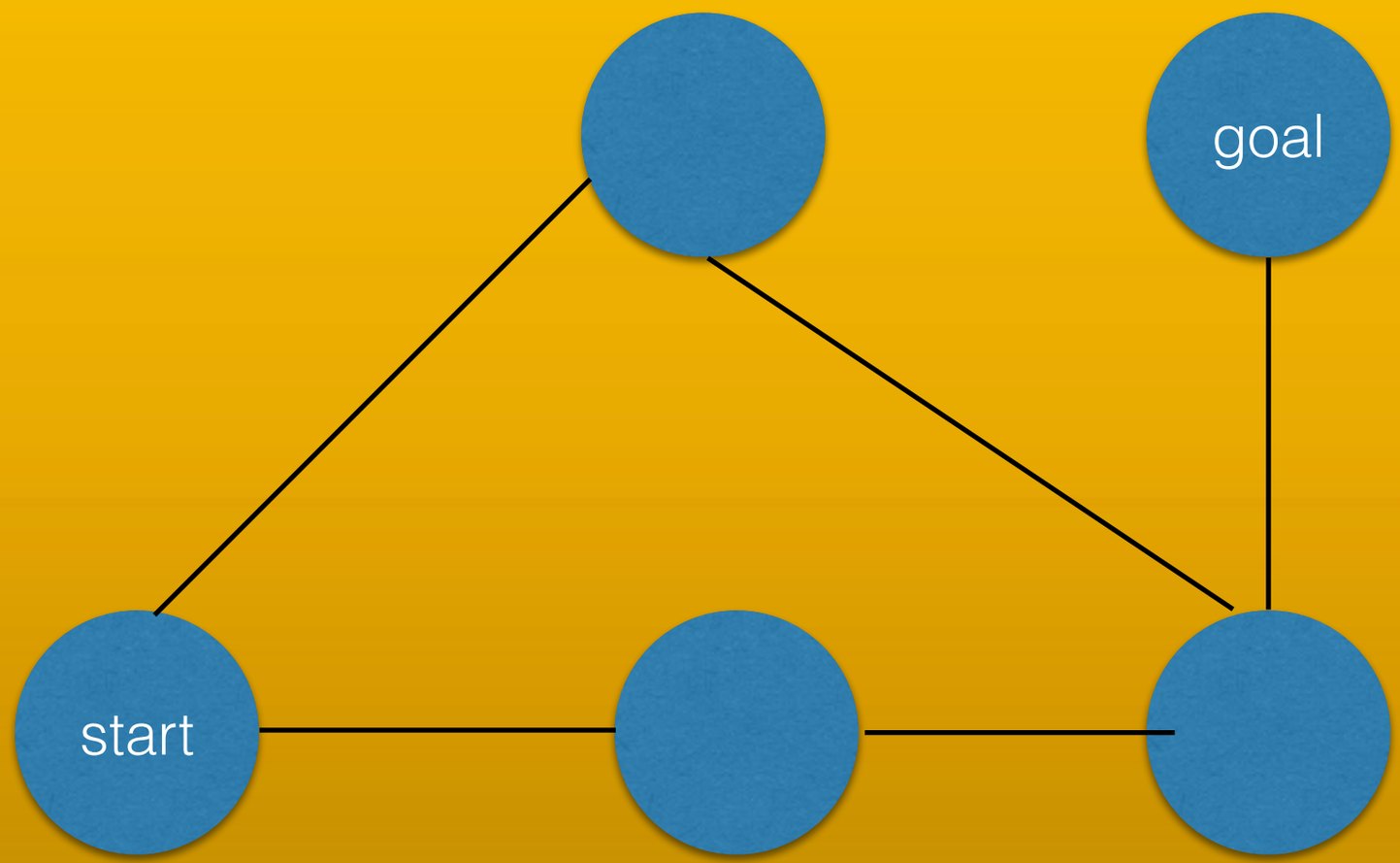
h(n)

let’s say for now, h(n) = straight line distance dari n ke Bucharest

Greedy Best First Search

$$f(n) = h(n)$$

gak optimal

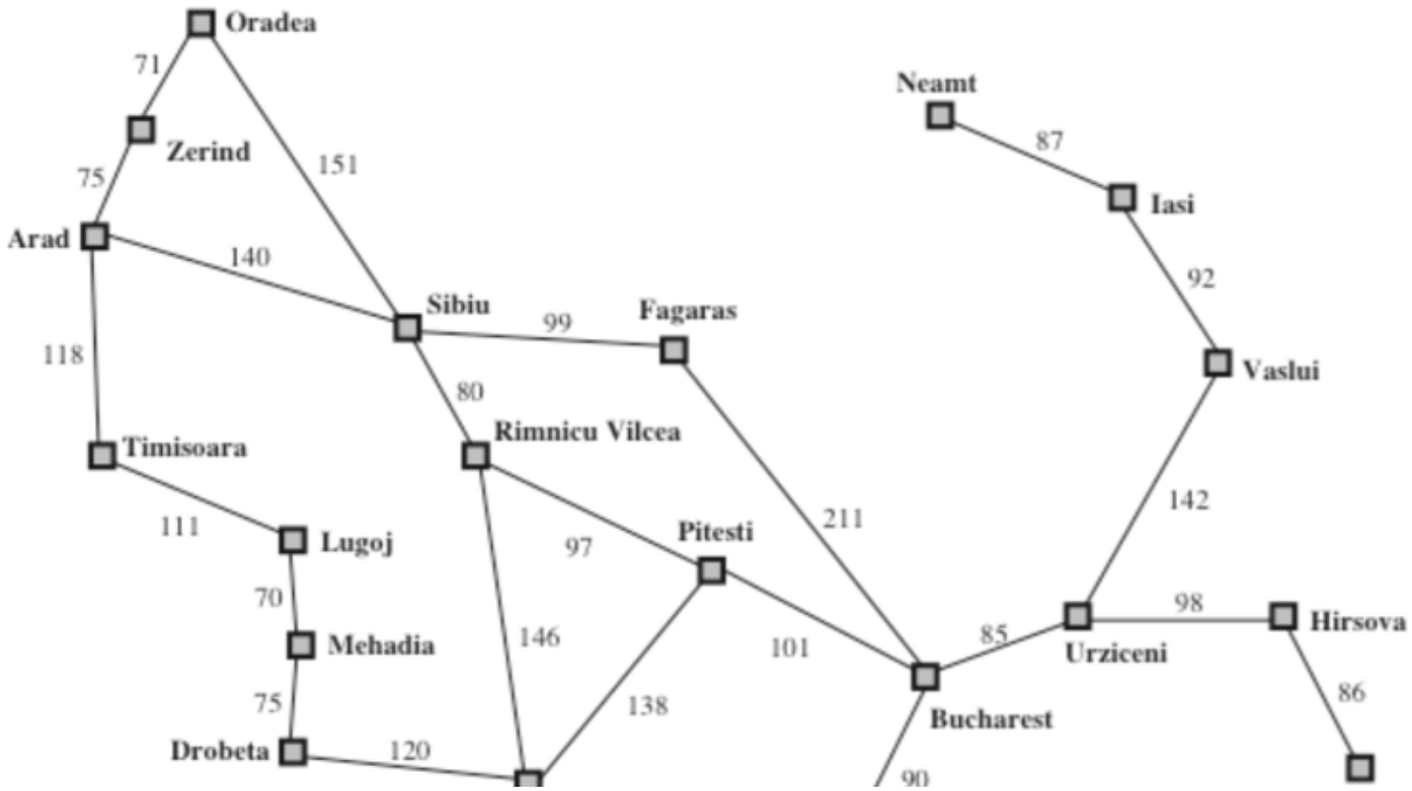


A* Search

ide : avoid path yang udah mahal

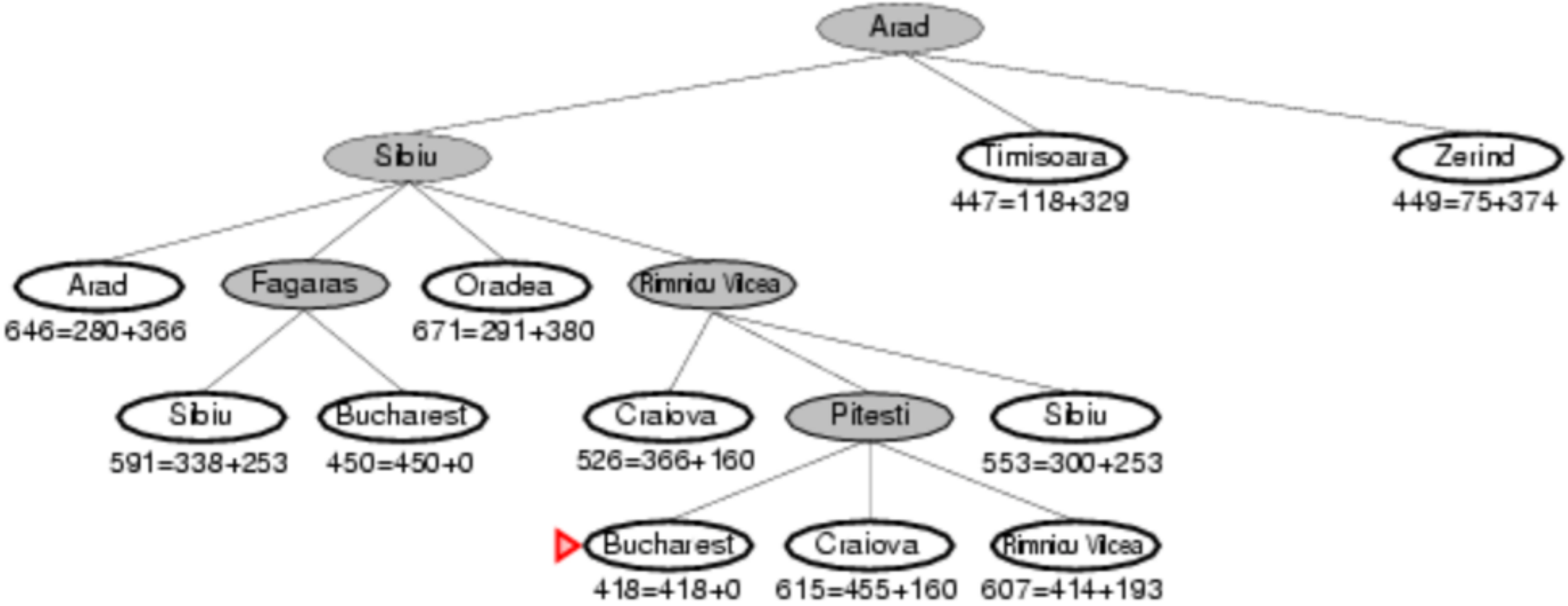
f(n) = g(n) + h(n)

g(n) = cost dari start ke n



Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199



gimana cari h(n) ?

h(n) itu **admissible** kalo dari semua node n, h(n) ≤ h*(n)

dimana h*(n) itu true-min-cost

contoh,
h(n) yang sebelumnya itu
admissible

intuitively, jalan belok2 gak akan lebih pendek daripada jalan lurus

why admissible?

$$f(x) = g(x) + h(x)$$

$$h(x) \leq h^*(x)$$

$$f(x) \leq g(x) + h^*(x)$$

why admissible?
idea : kalo lu pop sebuah finish node x dari pqueue, berarti semua element lain $f(x) \leq f(x')$.

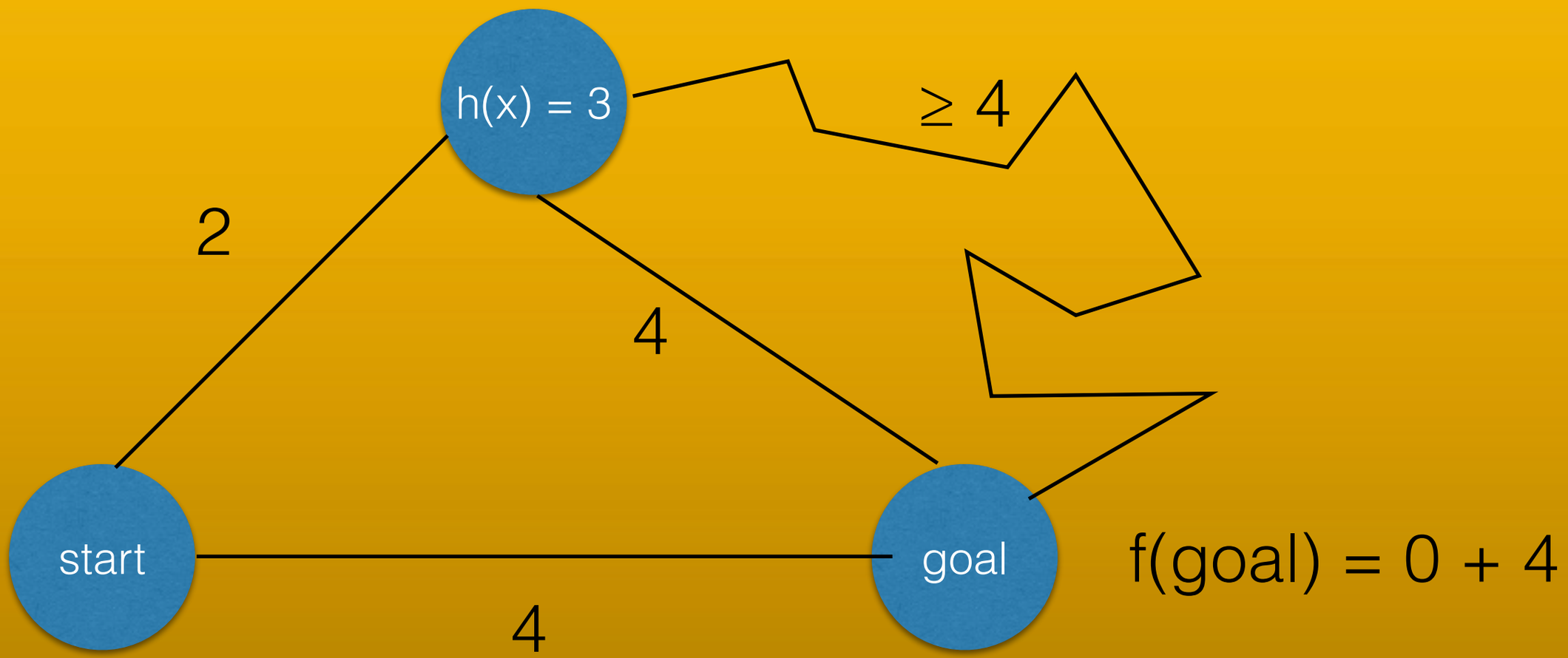
since $f(x') \leq g(x') + h^*(x')$

$f(x) \leq g(x') + h^*(x')$

maka x pasti optimal

contoh

f(x) = 2 + 3 = 5



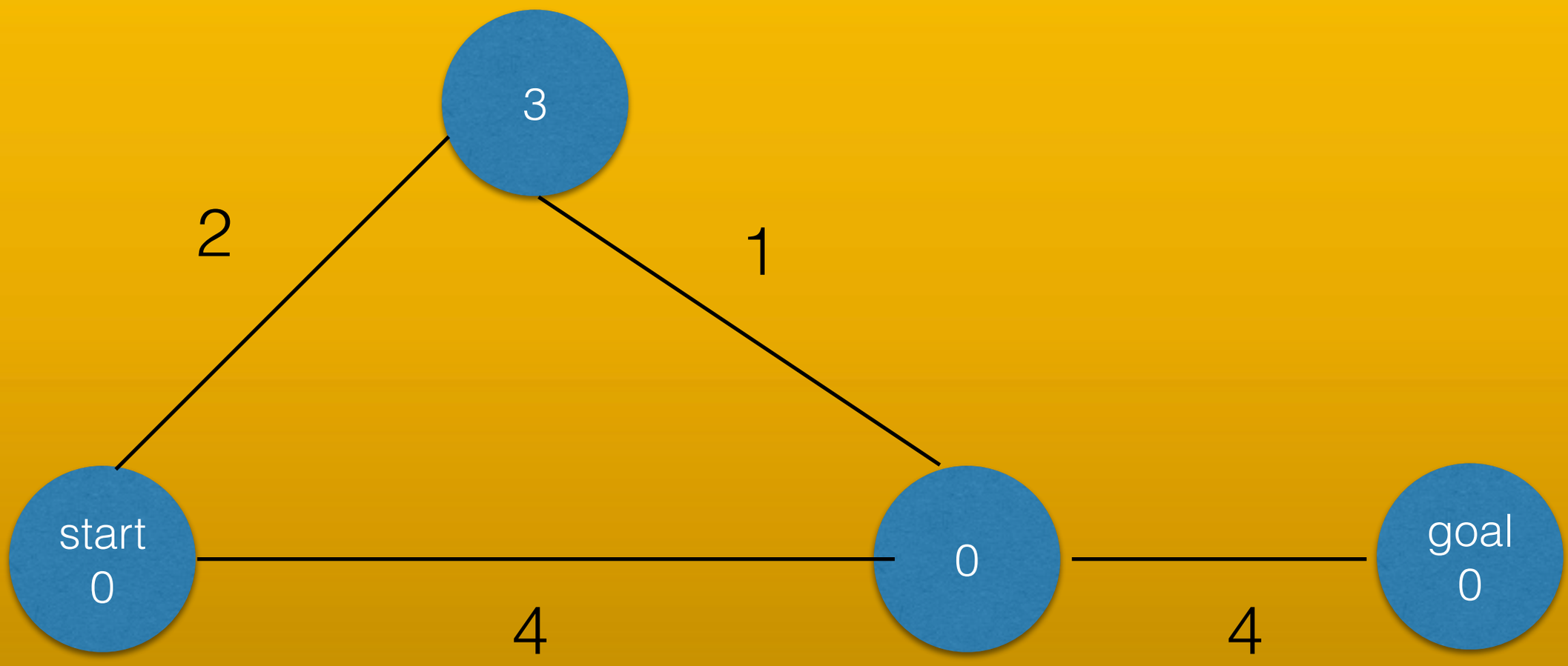
common mistake

admissible doang ga cukup

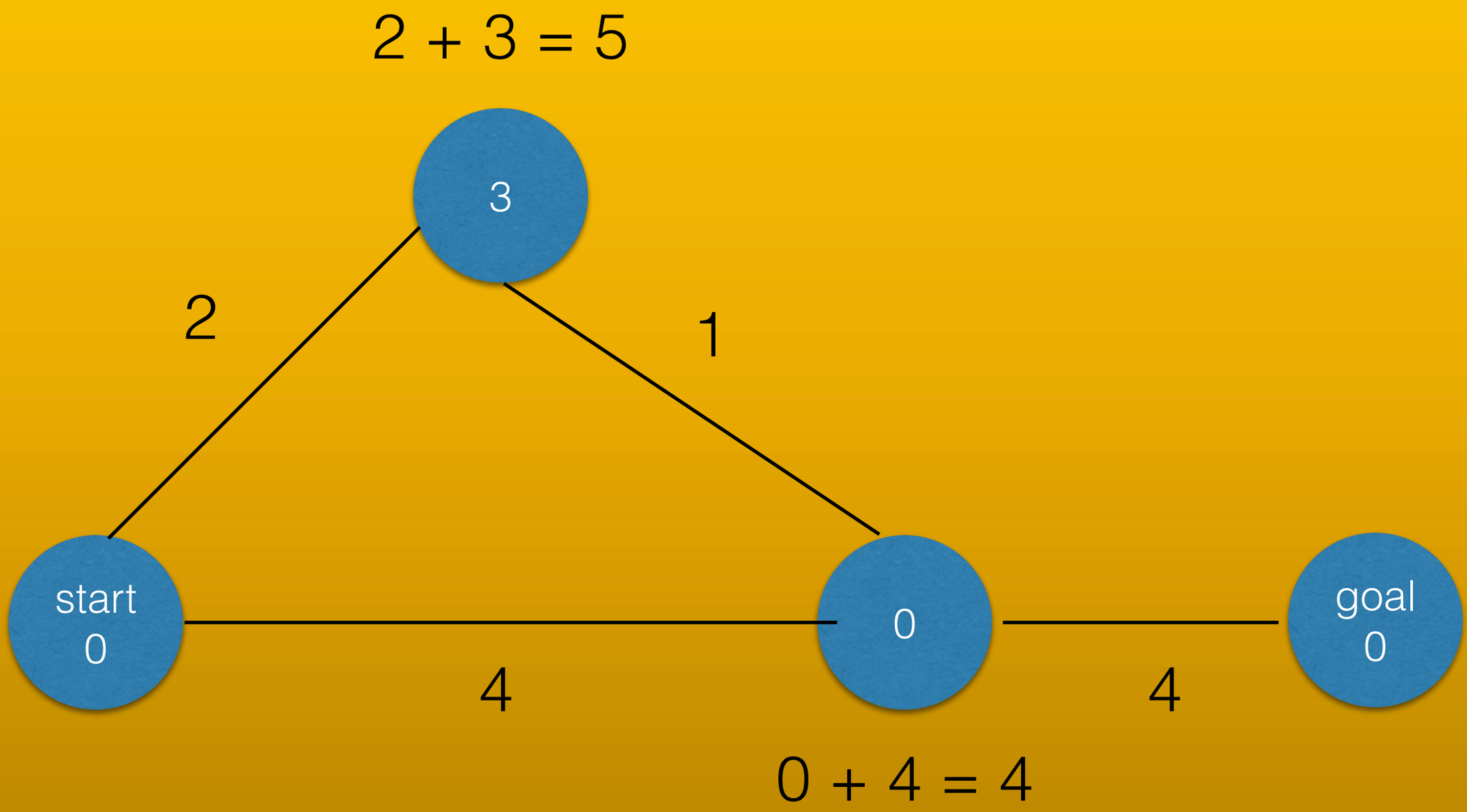
buat A*

gw “ditipu” pas pelatnas beberapa tahun lalu

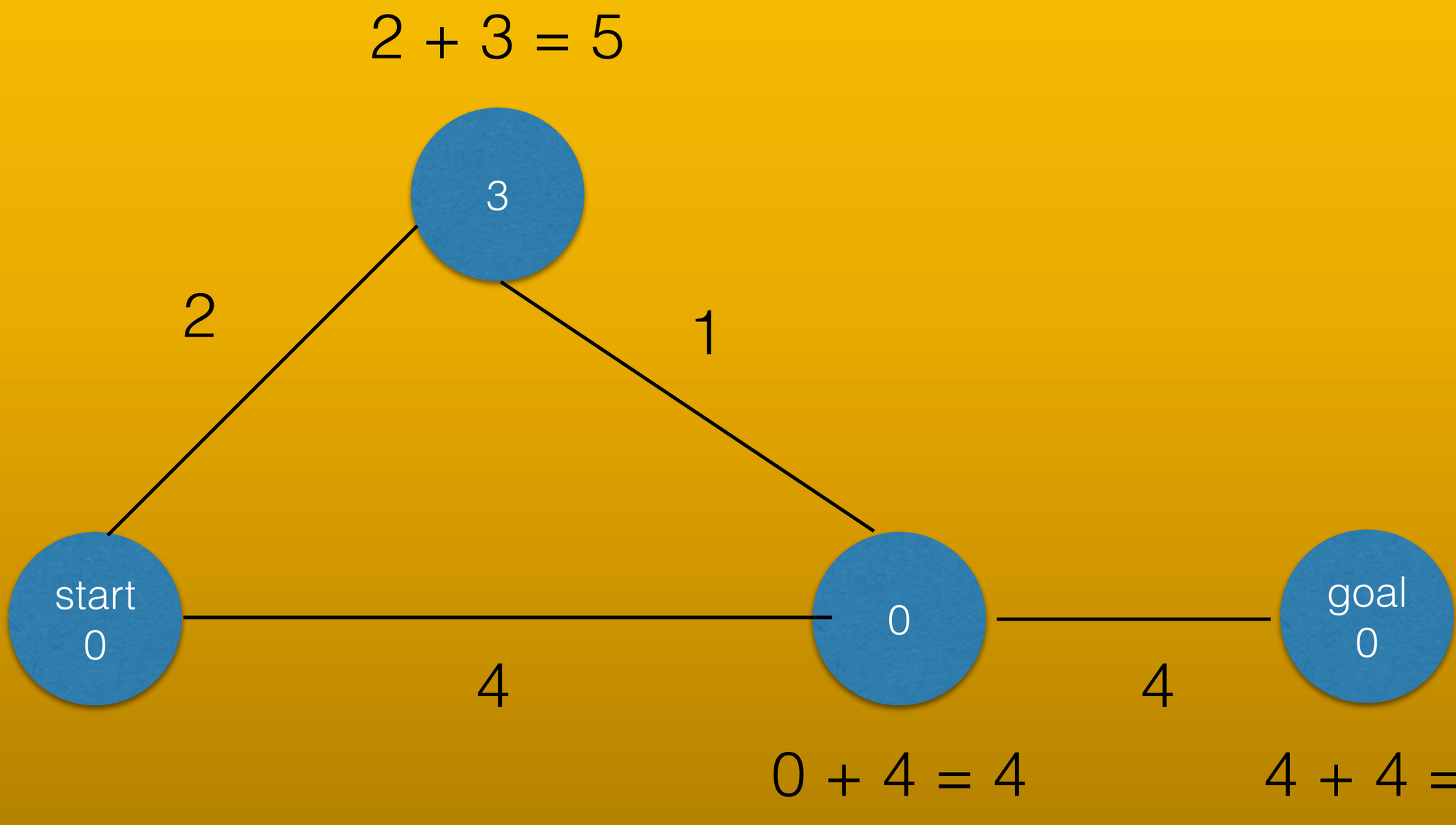
ini admissible, tapi gak optimal



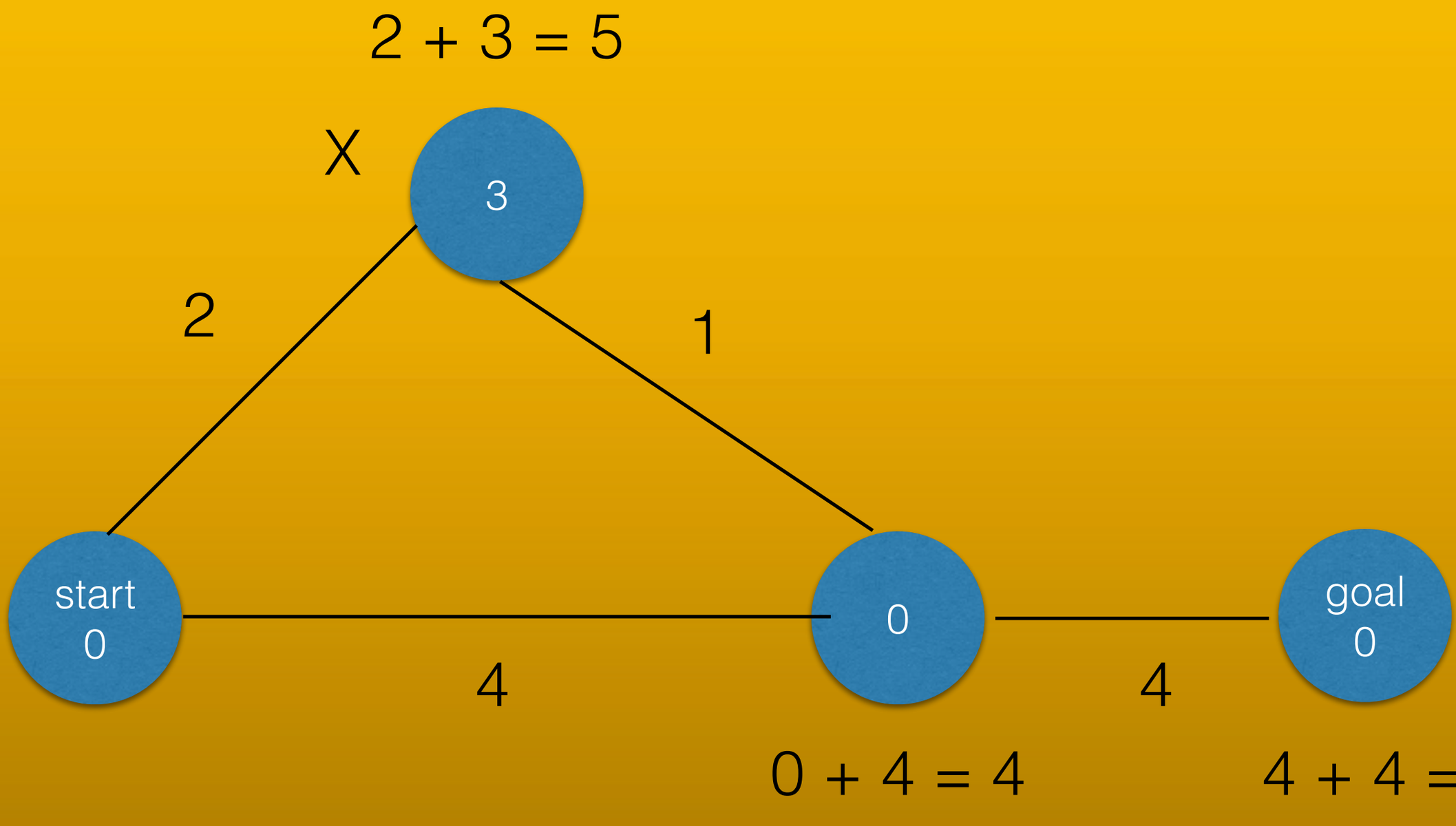
gak optimal



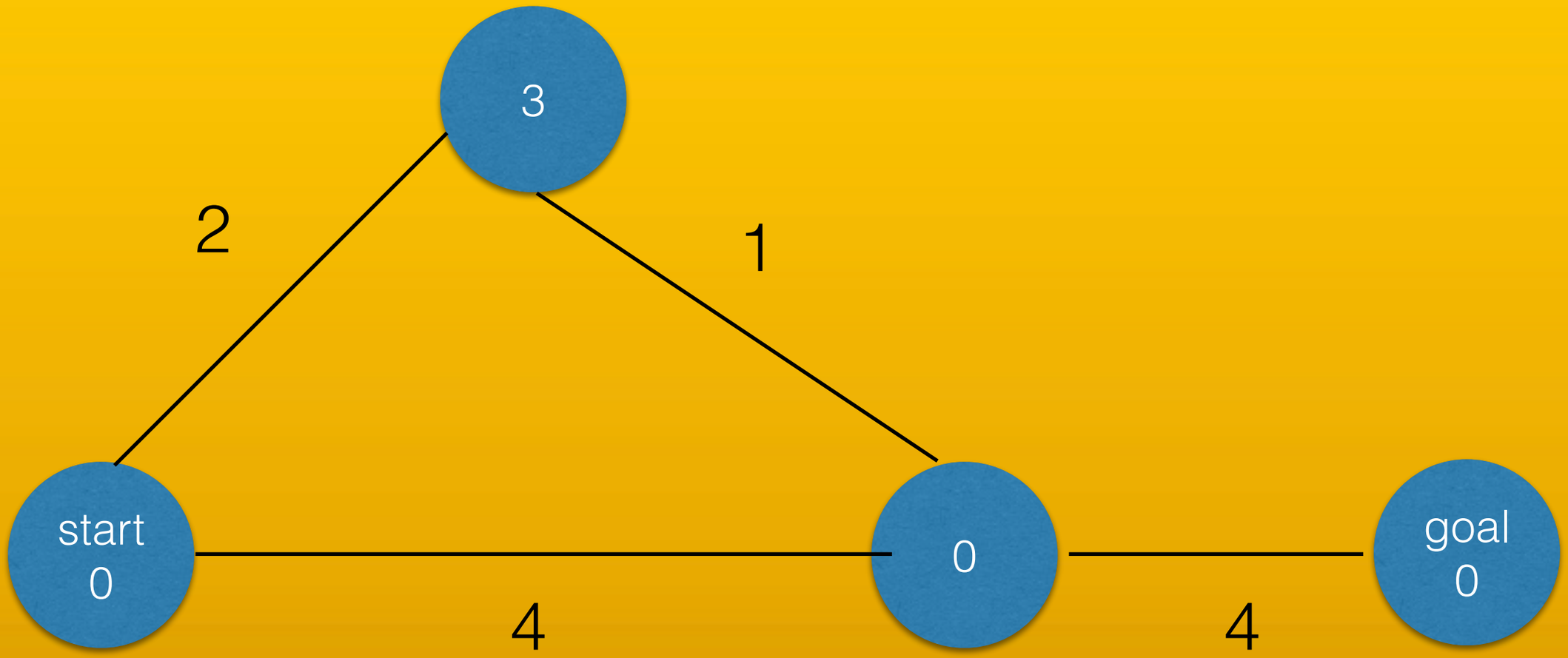
bakal pilih yang bawah duluan karena f(n) nya lebih kecil



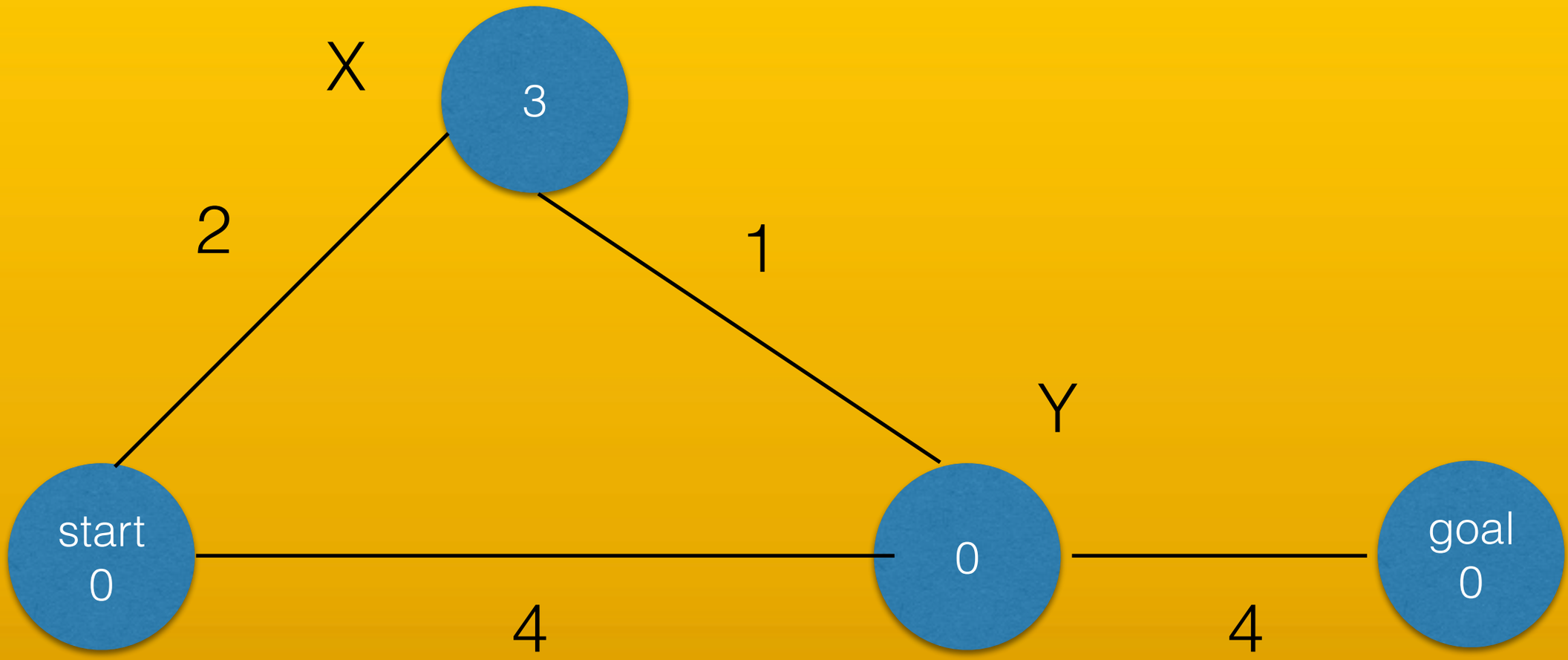
heuristic value X terlalu besar



h(n) itu **consistent** kalo dari semua node n dan n', h(n) ≤ h(n') + w(n,n')



ini gak consistent



$$h(X) = 3,\ h(Y) = 0,\ w(X,Y) = 1$$

$$h(X) > h(Y) + w(X,Y)$$

bosen ah teori mulu

mari kita masuk contoh2 yang aplikatif <3

contoh klasik

8-puzzle

	1	2
3	8	7
4	6	5

BFS

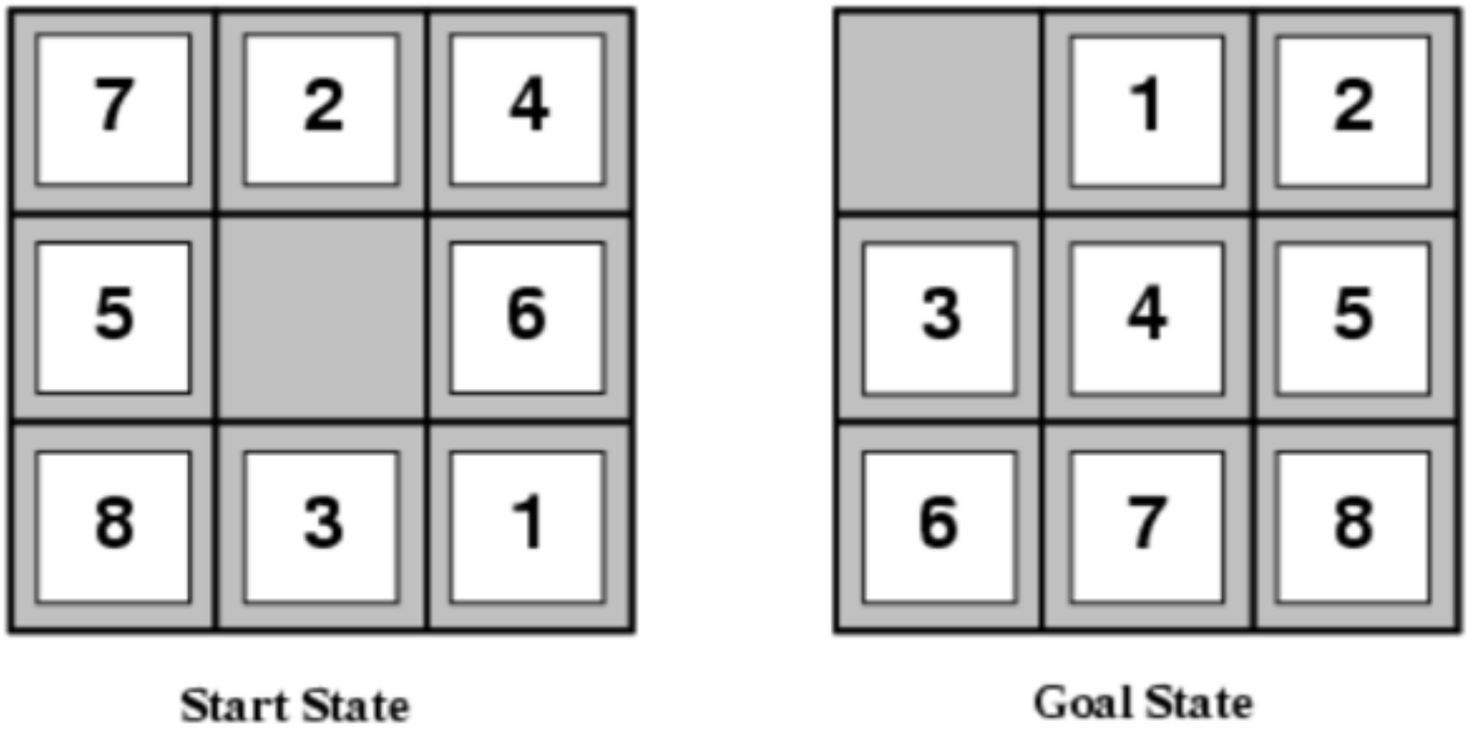
O(#state) = O(9!) = 362880

bisa sih, tapi coba kita solve pake A*

mari kita bikin h(n)

$h_1(n)$ = banyaknya tile salah tempat

$h_2(n)$ = total manhattan distance dari semua tile ke tempat sebenarnya



Start State

Goal State

- $h_1(S) = ?$ 8
- $h_2(S) = ?$ 3+1+2+2+2+3+3+2 = 18

suppose h_1 sama h_2 dua2 nya admissible, dan h2(n) ≥ h1(n) untuk semua n

kita bilang h2 dominates h1

h2 generally bakal incurs lower search cost than h1

gimana cari heuristic function yang admissible

game nya kita bikin relaxed

(e.g. tambah move allowed)

optimal cost dari relaxed problem pasti admissible

kalo rule 8-puzzle, kita ganti

piece yang ngisi kotak kosong

boleh dari piece manapun

optimal cost => h1

kalo rule 8-puzzle, kita ganti piece boleh gerak ke adjacent square yang mana aja => h2

latihan yu….

dikasih grid R*C. ada start position sama finish position. lu boleh gerak 4 arah. dibeberapa kotak ada obstacle

heuristicnya?

relaxed problem : anggap obstaclenya gak ada

h(n) = jarak manhattan dari n ke finish

note : algo search apapun (termasuk A*) bakal search semua node kalo finish position not found

in other words, bakal as bad as any search algorithm yang lu pelajari pas lu masih TK (DTK, BTK)

kalo di soalnya ada

`if there is no solution, print -1`

or sth like that which means gak guaranteed ada solusi

then it is **unlikely** that the solution is to make the search algorithm faster

unlikely ≠ never

IDA*

Iterative Deepening A*

duh gw ngantuk, beresin slide ini sampe jam 1 pagi

mirip sama IDS, tapi kita pake heuristic valuenya

kalo sebuah node n, f(n)
melebihi bound, fail. anggap
node n itu tidak ada

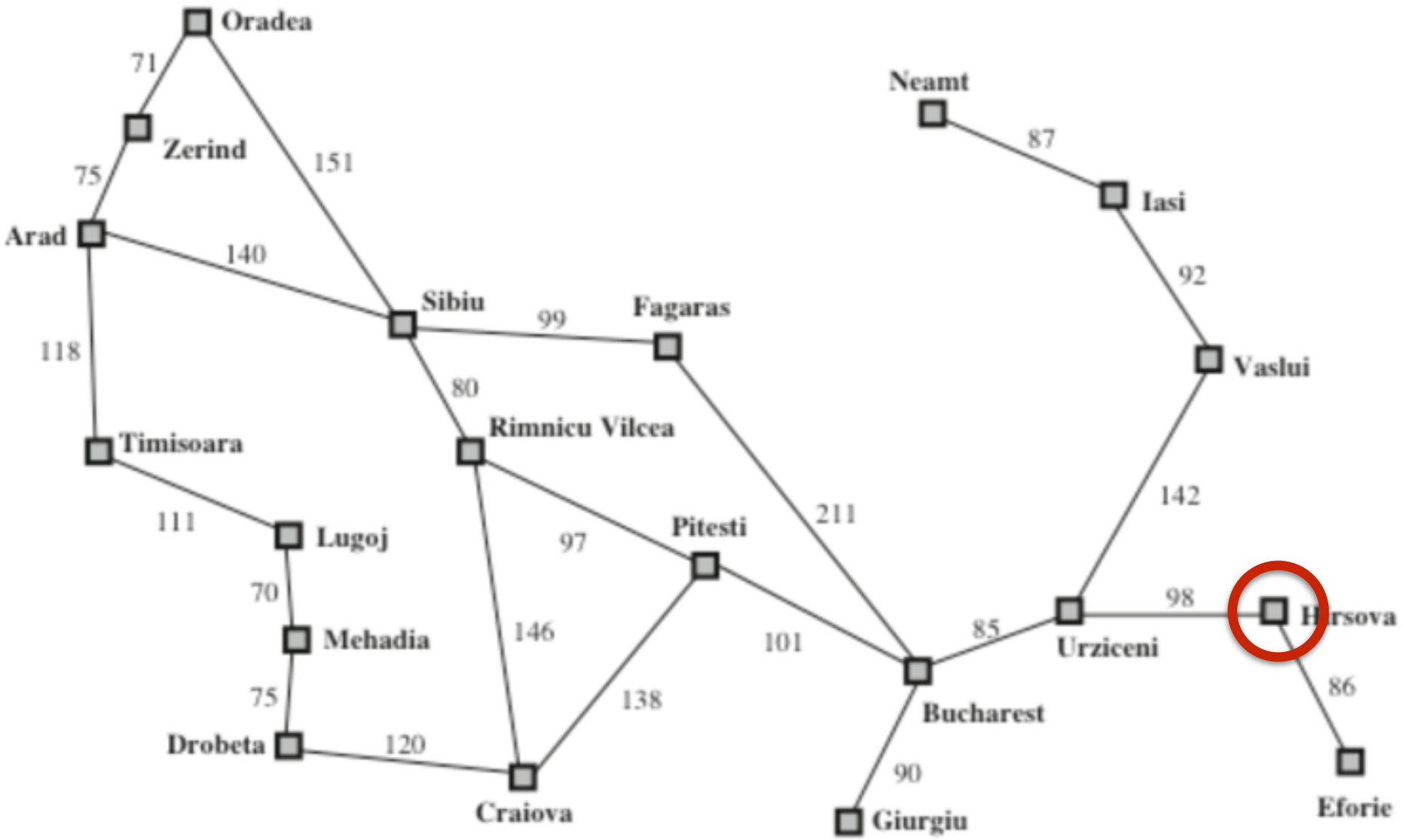
kalo finish found, return FOUND

otherwise, dari semua node n yang f(n) > bound, ambil f(n) terkecil dan jadiin bound baru

```
procedure ida_star(root)
   bound := h(root)
   loop
      t := search(root, 0, bound)
      if t = FOUND then return FOUND
      if t = ∞ then return NOT_FOUND
      bound := t
   end loop
end procedure

function search(node, g, bound)
   f := g + h(node)
   if f > bound then return f
   if is_goal(node) then return FOUND
   min := ∞
   for succ in successors(node) do
      t := search(succ, g + cost(node, succ), bound)
      if t = FOUND then return FOUND
      if t < min then min := t
   end for
   return min
end function
```

misal : Hirsova -> Bucharest
h(Hirsova) = 151, bound = 151

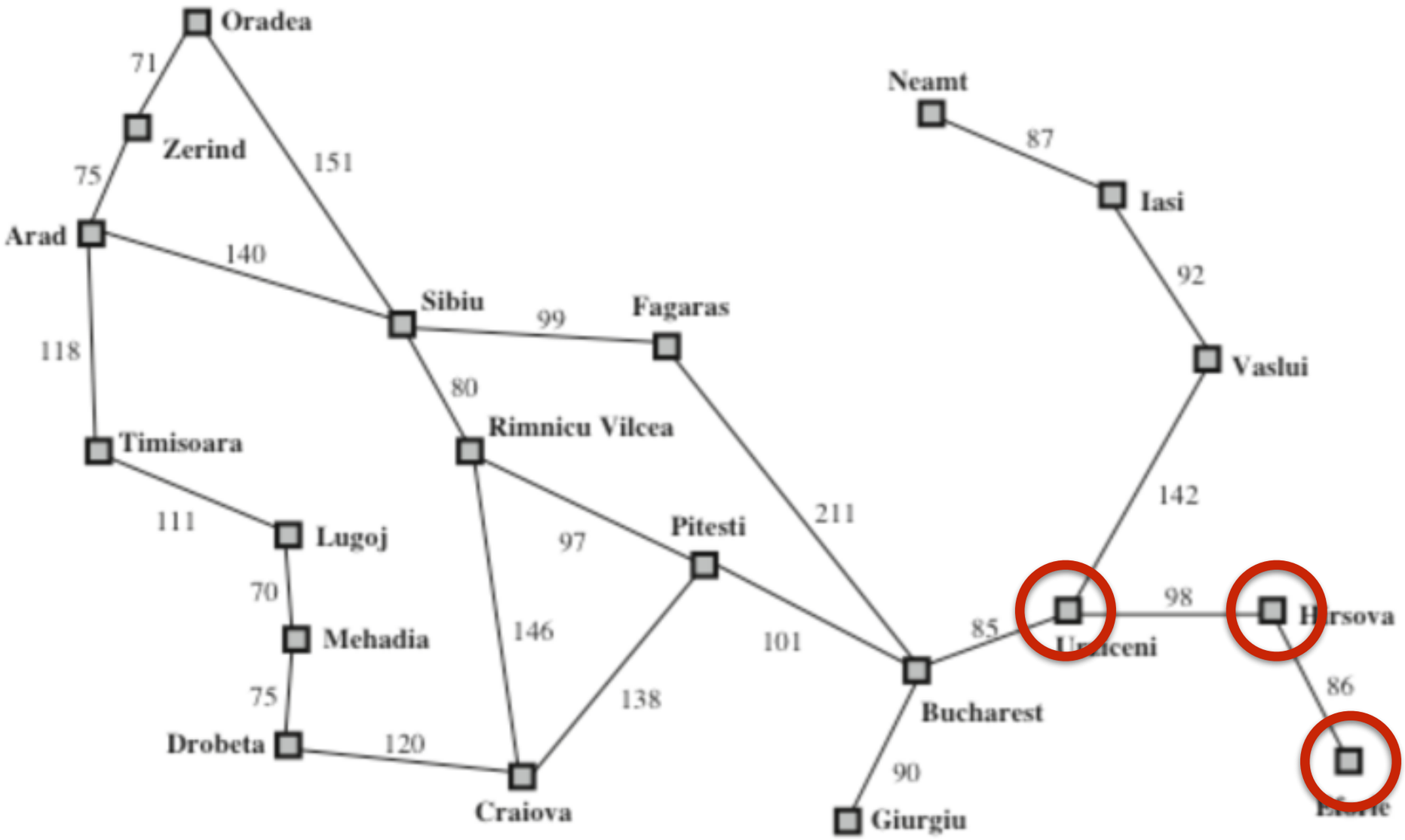


Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

f(Urziceni) = 80 + 98 = 178 - FAIL

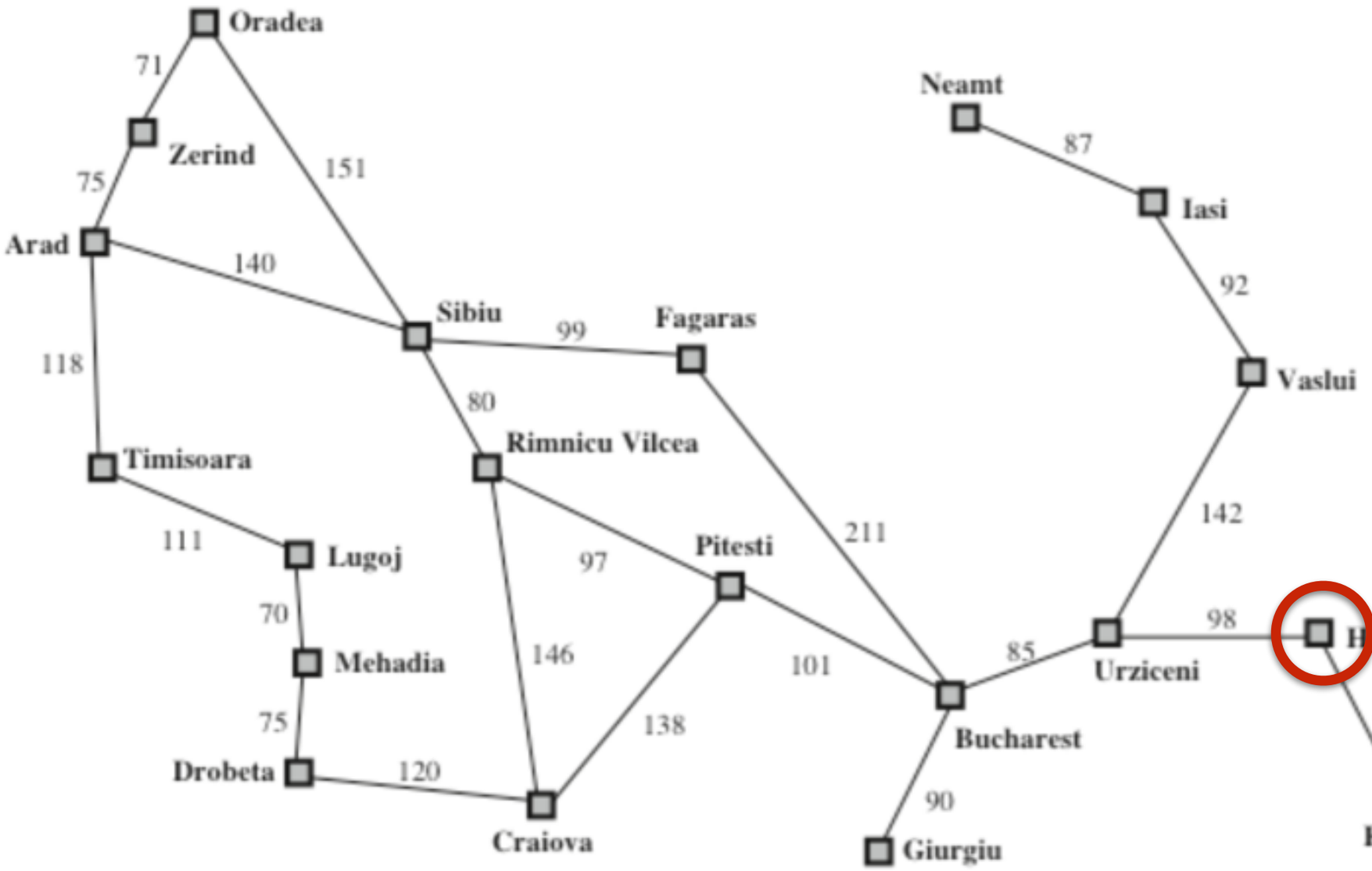
f(Eforie) = 161 + 86 = 247 - FAIL



Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

dua2nya fail, bound = 178
ulang dari awal

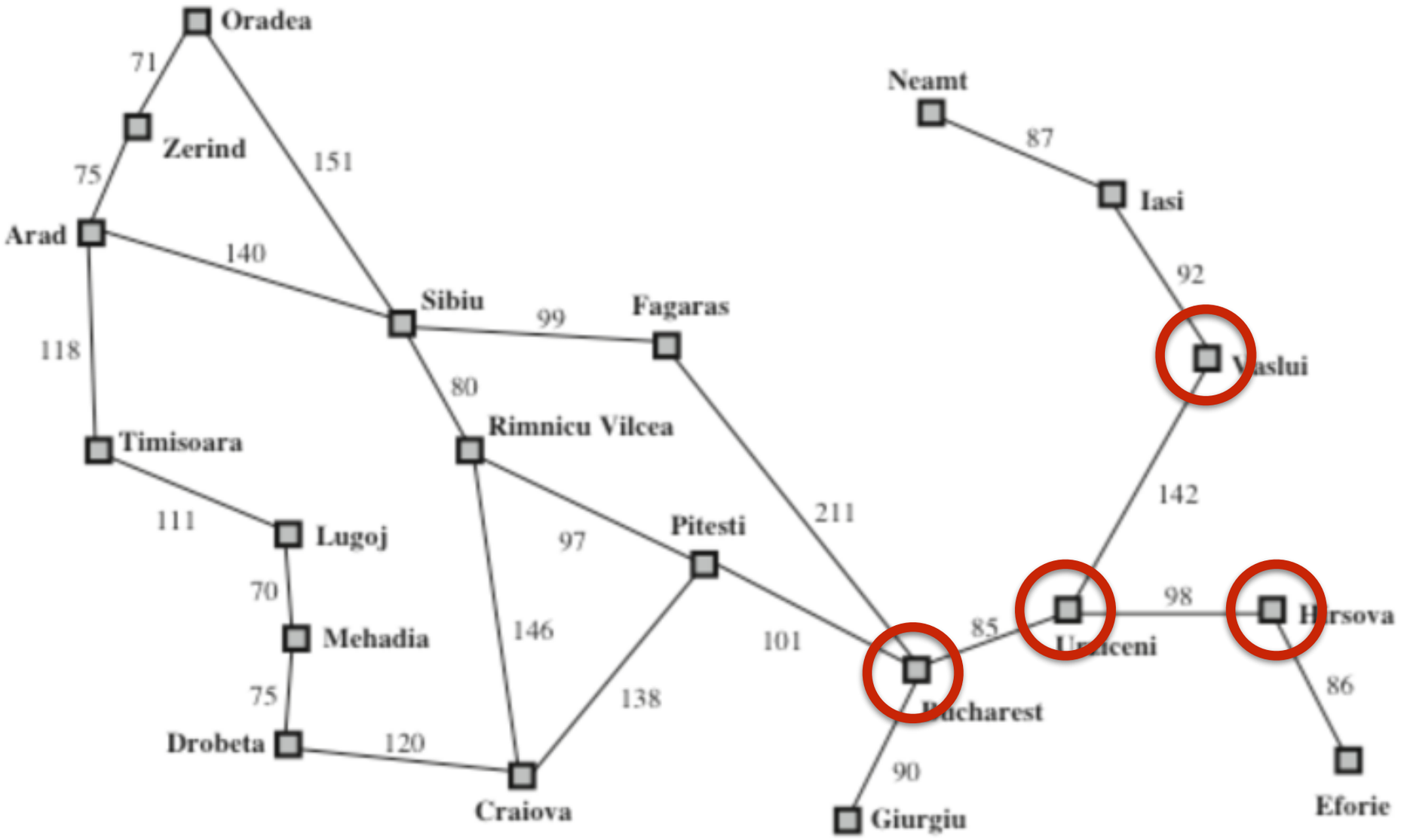


Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

f(Bucharest) = 0 + 85 + 98 = 183 - FAIL

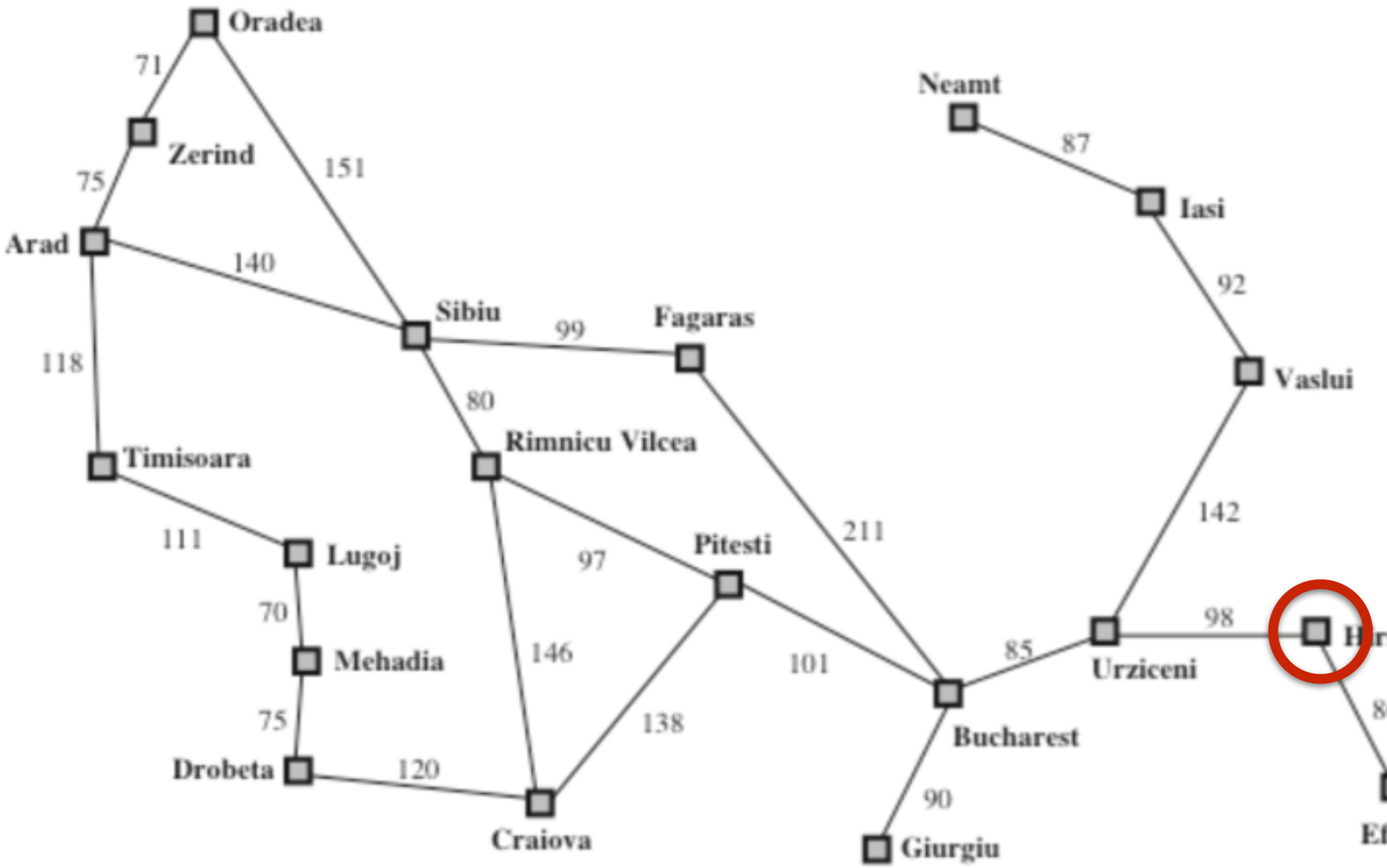
f(Vaslui) = 199 + 98 + 142 = 439 - FAIL



Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

bound = 183. ulang dari awal



Values of h_{SLD} - straight-line distances to Bucharest

Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

bound = 183 bakal dapet solusi
return 183

why IDA*?

EOF

Q&A?