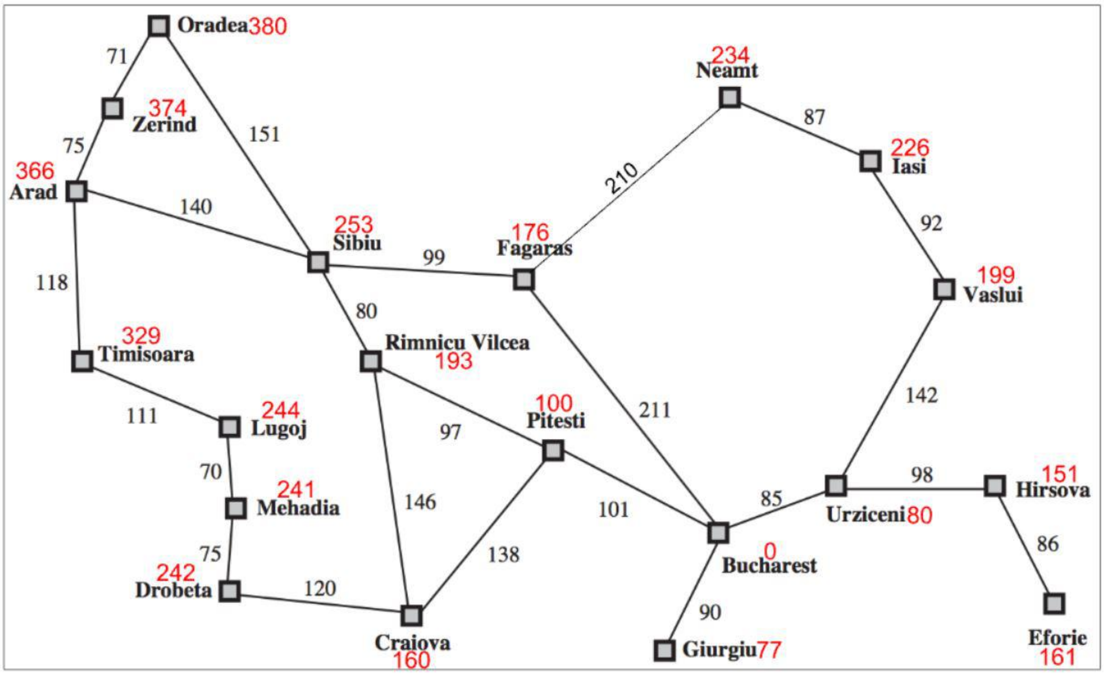
Zhuo Liu
CSCI-630-02 Foundations of FIS
Homework2



1 Using the Priority Queue

1) Start from Neamt
   h=234, g=0, f+h=234

2)
   Iasi which h=226, g=87, f=g+h=313
   Fagaras which h=176, g=210, f=g+h=386

   313<386, so choose Iasi

3)
   Vaskui which h=199, g=92+87=179, f=g+h=378
   Fagaras which f=386

   378<386, so we choose Vaskui

4)
   Urziceni which h=80, g=87+92+142=321, f=g+h=401
   Fagaras which f=386

   386<401, so we choose Fagaras

5)
   Sibiu which h=253, g=210+99=309, f=g+h=562
   Bucharest which h=0, g=210+211=421, f=g+h=421
   Urzicani which f=401

   401<421<562, so we choose Urzicani

6)
   Hirsova which h=151, g=98+142+92+87=419, f=g+h=570
   Sibiu which f=562
   Bucharest which f=421
   421<562<570, so we choose Bucharest
   Since Bucharest is the destination, so stop

Path=[Neamt, Iasi, Vaskui, Urzicani, Bucharest]

Total cost: 406

2 Using A* search for instance, the lowest path is show above. The heuristic function is corresponding to the decision tree we created. If the heuristic function will overestimate by at most k, the decision tree will change correspondingly as well. We can estimate the value of the nodes is each level of the decision tree.

3 a. hill climbing search

We can get the best state from each step

b. breadth first search

There is a successor function which contains all of the initial state

c. downhill move of hill climbing

d. random walk through search space

e. Random walk

Parents and children are same, but children with the possibilities of mutation

We will use a random successor

4 a. Assume the large rectangle has width W and height H. Each rectangle R_i is parameterized by four variables, x, y, w, h, which define its position, width and height. All smaller rectangles have the set of constraints

$$R_{i,w} \geq 0$$

$$R_{i,x} + R_{i,w} \leq W$$

$$R_{i,y} \geq 0$$

$$R_{i,y} + R_{i,h} \leq H$$

which confines each rectangle to the interior of the large rectangle. In addition, there is a set of constraints, C_{ij}, between rectangles i and j, $i \neq j$, which are

$$R_{i,x} + R_{i,w} \leq R_{j,x} \text{ or } R_{i,x} \geq R_{j,x} + R_{j,w}$$

$$R_{i,y} + R_{i,h} \leq R_{j,y} \text{ or } R_{i,y} \geq R_{j,y} + R_{j,h}$$

which enforces the constraint that no two rectangles overlap each other

b.

States: small rectangles which are in the larger one

Function: to place the small rectangle into the larger one

Tweak: we can use a BFS to search the large rectangle, and can use the set to store smaller rectangles which are already in the large rectangle

5 hill climbing:

Merit: we can modify the taste to a better one until we can get the best taste

Shortage: when the algorithm reaches Maximum and Minimum value, it’s hard to deal with and become random walk situation

Stochastic hill climbing:

Merit: we can make recipe randomly, when the better one shows, use the better one as the next state

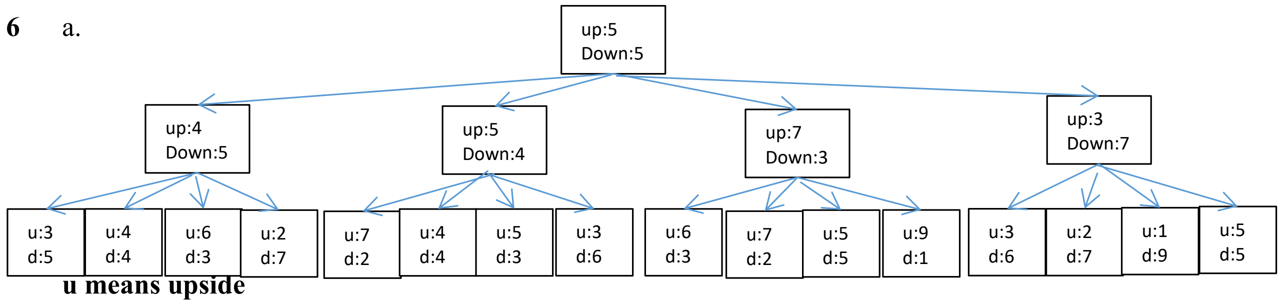
Shortage: unlike normal hill climbing, cannot find the maximum value

Genetic algorithms:

Merit: if better recipe shows up we need to do crossover and may do it recursively, and because of its flexibility we can generate random and versatile outcomes

Shortage: highly rely on parents

6 a.



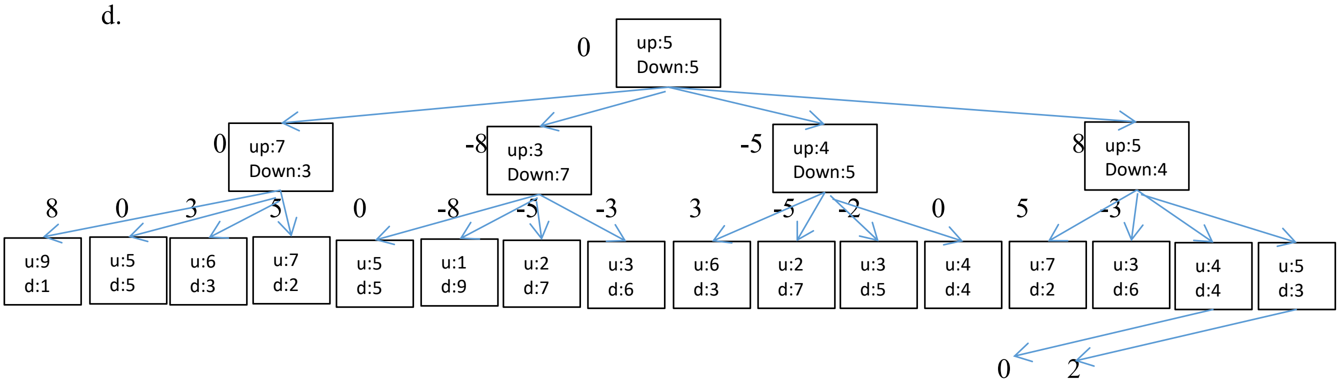
u means upside

d means downside

b. branch factor would be 4

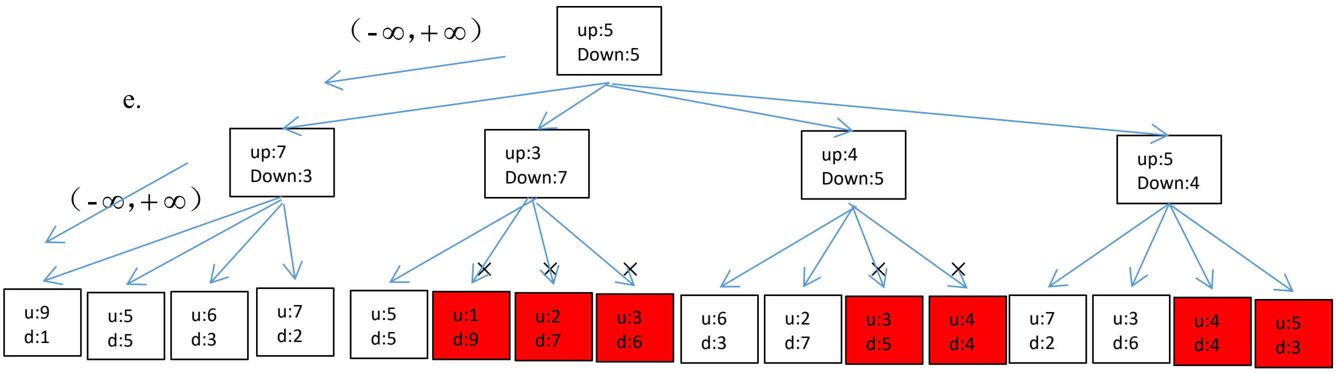
c. the depth of tree created in Problem 6.a should be 2 and the depth of this game tree should be infinity (in one situation one player flips two coins and the other one flip these two back)

d.



The non leaf node value is 0,-8,-5,-3

e.



Red color represents the nodes that would not need to be evaluated

7 State space : nine balls or 22 balls(snooker) and the location of balls and table, player

Possible actions: The objective of the game of snooker is to strike the white *cue ball* with a cue so that it strikes the *object balls* in turn and causes them to fall into one of the six pockets. Points are scored for potting balls legally, in accordance with the rules described below, or in the event of a foul committed by the opponent. The player who scores more points wins the frame, and the first player to win a set number of frames wins the match.

Difference: snooker can make as many as moves as possible(depend on the snooker's game rule) but according to the chess rule, one player can only make one movement each time

Snooker has infinite branching factor, because the route of the ball depends, but if using artificial intelligence, the branching factor of chess is predictable