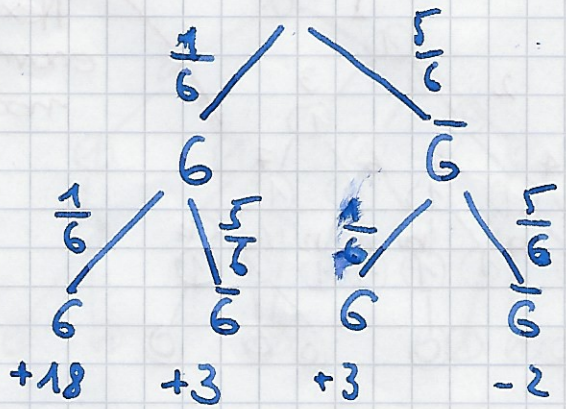
AI - Assignment 2

Lukas Graber: 4117151
Dorotea Lleshaj

Question 1: Rationality

Q_1	Q_2	Q_3	$\mathbb{E}$
5	4	10	19

a)



Expected income per round:

$$E(X) = \frac{1}{36} \cdot 18 + \left(\frac{5}{36} + \frac{5}{36}\right) \cdot 3 + \left(\frac{25}{36} \cdot (-2)\right)$$

$$= \frac{18}{36} + \frac{30}{36} - \frac{50}{36}$$

$$= \frac{48}{36} - \frac{50}{36} = -\frac{2}{36}$$

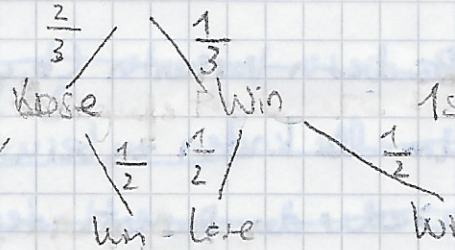
⇒ The expected income per round is negative. That means on average you will lose money. So don't play this game.

b) First version with 3 cups:

$$E(X) = \frac{1}{3} \cdot 6 - \frac{2}{3} \cdot 4 = \frac{6}{3} - \frac{8}{3} = -\frac{2}{3}$$

⇒ The expected income is negative. That means you lose money on average. Don't play this game.

Second version:



2nd choice is not random, switching ~~also~~ give $\frac{2}{3}$ chance of winning

$$\left(\frac{2}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2}\right) \cdot 6 + \left(\frac{2}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2}\right) \cdot (-4)$$

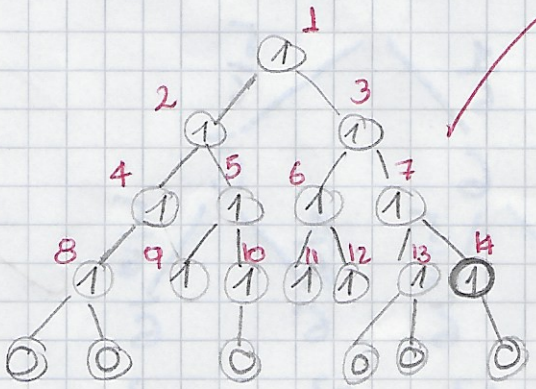
$$= \left(\frac{2}{6} + \frac{1}{6}\right) \cdot 6 + \left(\frac{2}{6} + \frac{1}{6}\right) \cdot (-4) = \frac{3}{6} \cdot 6 - \frac{3}{6} \cdot 4 = 3 - 2 = 1$$

⇒ The expected income is positive. That means you should play this game. The new version improves the chances to win this game.

Question 2 - Search strategies:

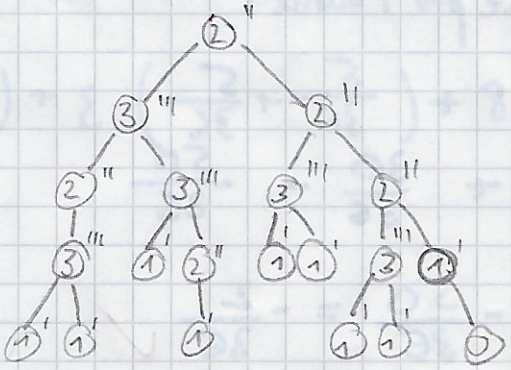
4/4

a) Breadth-first search

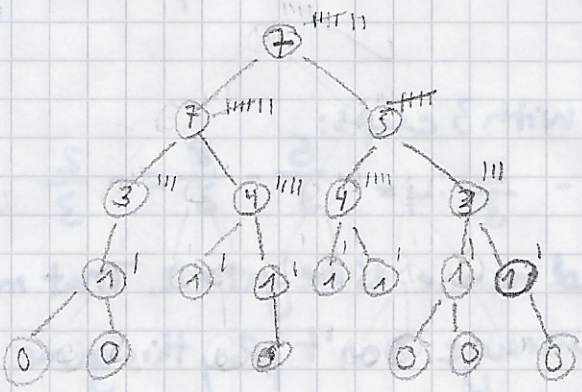


Next time enumerate the nodes.

b) Depth-first search:



c) Iterative deepening depth-first search:

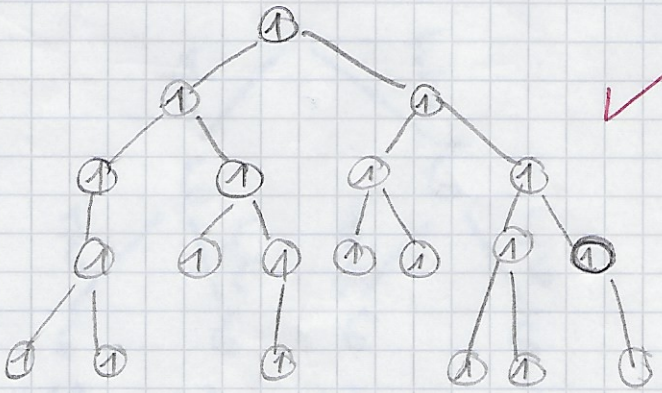


=> ~~Bei diesen Varianten (b, c) wird Backtracking verwendet. Um alle Knoten zu besuchen muss man also potenziell immer wieder den Parentknoten erreichen können. Da wir uns nicht sicher sind~~

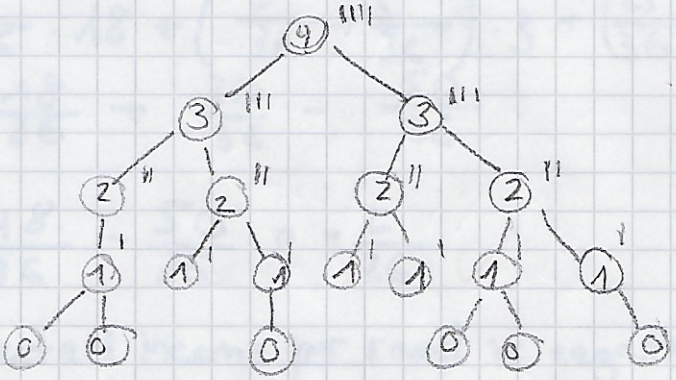
These solutions (b, c) uses backtracking to reach every node. It is the way we learned DF search in prior lectures. So for safety reasons, we thought we should at least mention them. On the other page, you can find the solution with the stack.

Lukas Graber

b) Depth-First Search with Stack (Slides p. 28) Dorcka Lleshaj



c) Iterative Deepening Search with Stack (Slides p. 33):



=> ~~Bei diesen Varianten (b, c) wird ein Stack wie auf den Folien verwendet. Dann kann man sich viele Besuche sparen und trotzdem alle Knoten besuchen.~~

=> These solutions (b, c) use a stack like in the slides. By doing so, you can save most of the visits of parent nodes. You basically don't have to backtrack.

Question 3:

Source code to be found in Ilias.