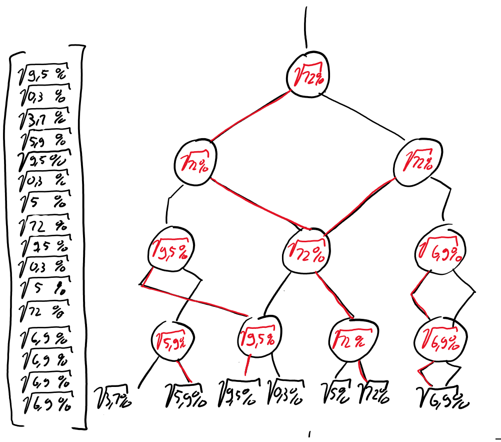
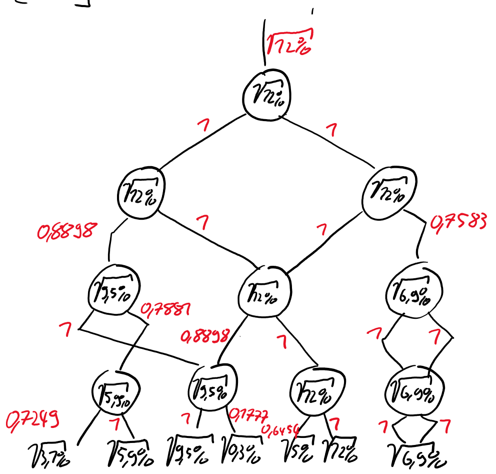
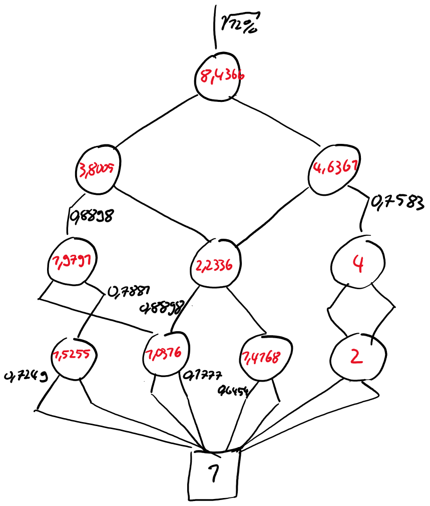
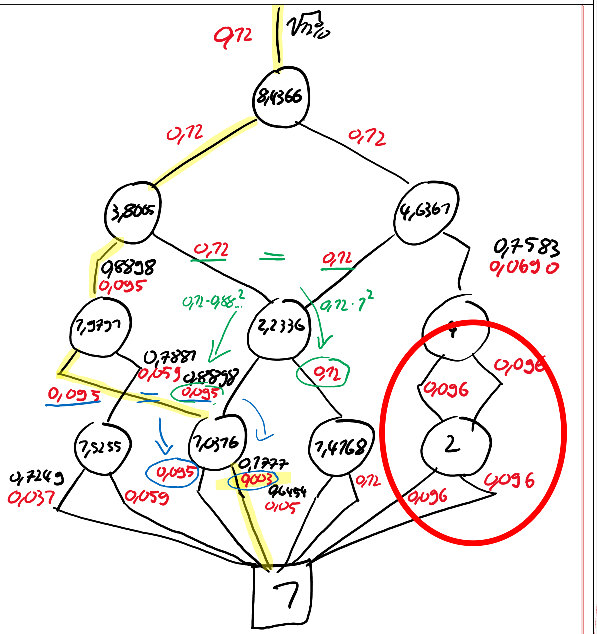
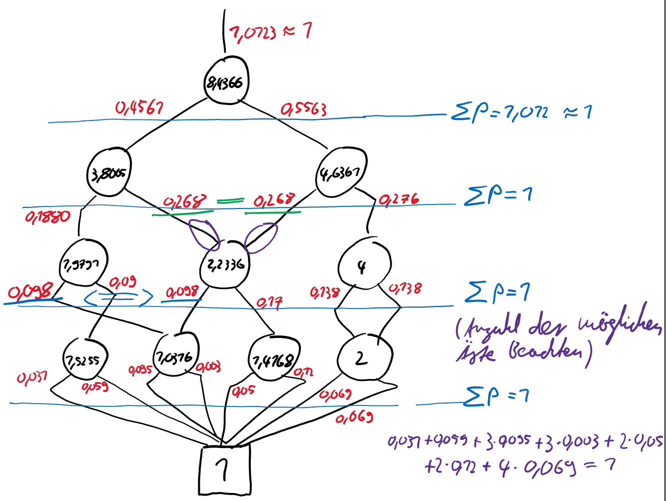
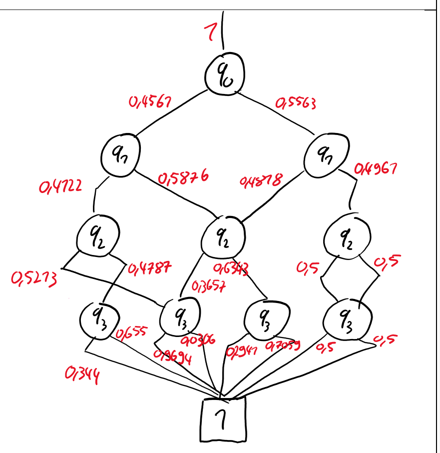
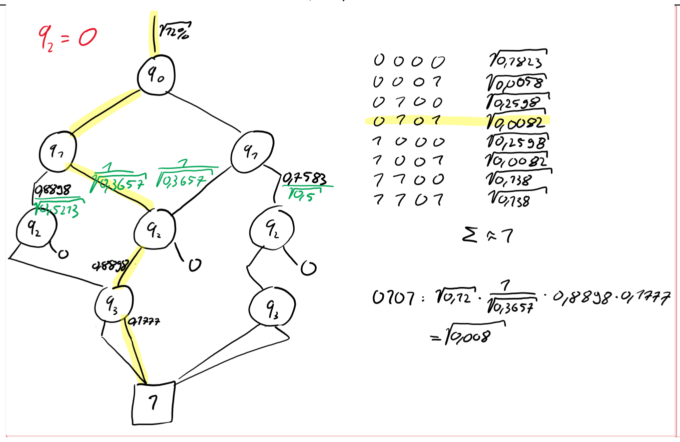
<table>
<tr><td colspan="2">1. Erstelle Objekt für DecisionDiagram mit Matrix/Vektor zum Initialisieren</td></tr>
<tr><td colspan="2">2. Untersuche Matrix/Vektor auf gleiche Teilmatrizen/Teilvektoren (Gleiche Knoten speichern / DD vereinfachen)</td></tr>
<tr><td>3. Unten beginnend: Für alle Knoten den betragsmäßig größten Wert der Nachfolgeknoten auf den betrachteten Knoten übernehmen (In der letzten Ebene sind die Nachfolge-Werte die Einträge des Zustandsvektors)

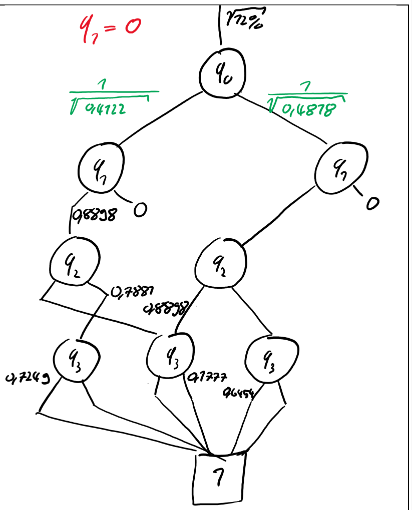
Vielleicht würde es mehr Sinn machen, den kleineren Wert zu nehmen? Höhere Genauigkeit</td><td></td></tr>
<tr><td>4. Kantengewichte bestimmen: Wert des Nachfolgeknoten dividiert durch den Wert des Quellknoten (betrachtete Kante ist zwischen diesen Knoten)</td><td>
</td></tr>
<tr><td>5. Die Werte der letzten Ebene (Elemente des Zustandsvektors) auf 1 setzen, wenn Eintrag ungleich 0 war (oder auf 0 setzen, wenn Eintrag 0 war).
6. Knoten neu zusammenfassen (unten beginnend: Knoten mit selben Kantengewichten zusammenfassen oder wenn gespeicherter Wert gleich ist (rot))</td><td rowspan="2">
</td></tr>
<tr><td>7. Die „gewichtete Wahrscheinlichkeit“ berechnen: $p = p_{links} \cdot \omega_l^2 + p_{rechts} \cdot \omega_r^2$ (Wert des linken Nachfolgeknoten multipliziert mit dem quadrierten Kantengewicht der linken Kante, addiert mit derselben Rechnung für die</td></tr>
</table>

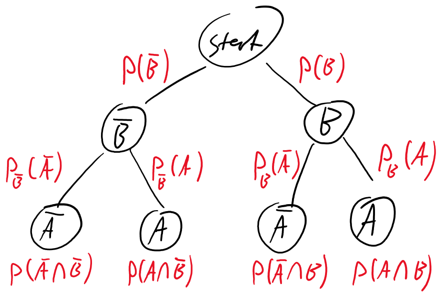
rechte Kante)	
8. Für alle Kanten das Produkt der Kantengewichte auf dem jeweiligen Ast berechnen (von der betrachteten Kante bis zur Kante über dem Wurzelknoten). Bei mehreren eingehenden Kanten muss dieses Produkt bei allen eingehenden Kanten gleich sein, damit man mit diesem Wert an der betrachteten Kante weiterrechnen kann (blau und grün). → Sicherheitshalber in Rechnung prüfen oder Beweis finden. Sonst müsste Knoten an dieser Stelle auf zwei Knoten aufgeteilt werden (oder erst am Ende vereinfachen). Werden Knoten nach Kantengewichten zusammengefasst, sind diese Werte nicht immer gleich. Programmierung muss anders erfolgen. (Anstatt Multiplikation mit Anzahl wie oft Kante vorkommt, Addition mit der Wahrscheinlichkeit für die Kante in allen möglichen Ästen, in der die Kante vorkommt.)	
9. Für jede Kante wird dieses Produkt mit dem im Knoten, am unteren Ende der jeweiligen Kante, gespeichertem Wert multipliziert. (Nach Multiplikation mit der Anzahl mit der eine Kante in allen Ästen vorkommt, ergibt sich die Wahrscheinlichkeit, dass diese Kante durchlaufen wird.) Anzahl der Möglichkeiten wird jetzt nicht mehr benötigt, durch Addition im vorherigen Schritt erfolgt dies automatisch. Man hat also direkt die Wahrscheinlichkeit für diese Kante gegeben, man hat aber keine Information darüber, in welchem Ast genau die Kante enthalten ist → mehrere Äste enthalten diese Kante (Wahrscheinlichkeiten für Kante in einem bestimmten Ast vorkommend, ist jetzt unterschiedlich) (Elementare Wahrscheinlichkeit, z.B. $P(0 \cap 1 \cap 0 \cap 1)$)	 Wahrscheinlichkeit, dass eine Kante durchlaufen wird: $P(A)$, $P(A \cap B)$, $P(A \cap B \cap C)$ ...

Kommentiert [LL1]: Fehler: 0,069 statt 0,096!!!

Wird **doch** benötigt: Bestimmung der bedingten Wahrscheinlichkeiten (Entscheidung: entweder links oder rechts): $P_{01}(q_2 = 1) = \frac{P(0\cap1\cap1)}{P_0(q_1=1)}$)	 bedingte Wahrscheinlichkeiten $P_A(B)$
10. Messung durchführen: Zufallszahl zwischen 0 und 1 erzeugen, ist diese kleiner gleich p_{links} / $P_{010}(q_3 = 0)$ wird das jeweiliges Qubit mit 0 gemessen. Andernfalls mit 1. Messung anhand der summierten Wahrscheinlichkeit q_2=0. Es ergeben sich mehrere Zustände (Vektor) Alternativ könnte bei anderer Vorgehensweise in den oberen Schritten, wo Knoten nicht zusammengefasst sind und Informationen über Ast indem Kante enthalten ist vorhanden sind, ein einzelner Zustand gemessen werden. Das entspricht dann der Messung aus dem Eintrag des ursprünglichen Zustandsvektors…	
11. Die nichtgemessene Wahrscheinlichkeit wird zu 0, der darunterliegende Baum fällt weg.	

Kommentiert [LL2]: Wahrscheinlichkeiten der eingehenden Kanten können unterschiedlich sein, deswegen besser ausgehende Kante normieren

12. Entscheidungsdiagramm neu normieren: Auf der Ebene, wo das Qubit gemessen wurde, werden die eingehenden Kanten aller Knoten dieser Ebene durch die Wurzel der Wahrscheinlichkeit geteilt, mit der das Ergebnis gemessen wurde. Diese kann für jeden Knoten anders sein.	
13. Für die Einträge im Zustandsvektor müssen alle Kantengewichte eines Astes für alle möglichen Äste multipliziert werden. Die Entscheidung 0, 1 muss sich gemerkt werden → jeder Ast wird durch eine binäre Zahl dargestellt. Umwandlung in Dezimalzahl gibt Index des Eintrags im Zustandsvektor an.	



$$P(A \cap B) = P(B) \cdot P_B(A) = P(A) \cdot P_A(B)$$

$$P_B(A) = \frac{P(A \cap B)}{P(B)}, \quad P_B(\bar{A}) = \frac{P(\bar{A} \cap B)}{P(B)}, \ldots$$

$$P_B(A) + P_B(\bar{A}) = 1$$

$$P(A) \cdot P_A(B) = P(B) \cdot P_B(A)$$

$$P(A) = P(B) \cdot P_B(A) + P(\bar{B}) \cdot P_{\bar{B}}(A)$$

$$P_A(B) = \frac{P(B) \cdot P_B(A)}{P(B) \cdot P_B(A) + P(\bar{B}) \cdot P_{\bar{B}}(A)}$$