

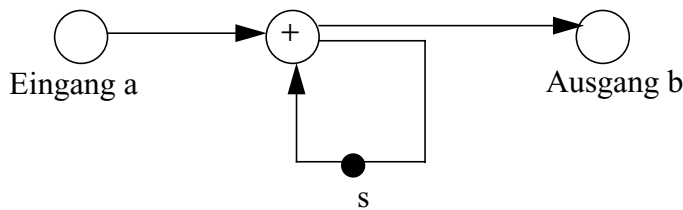
Aufgabe 1: Spezifikation und Modellierung

(35 Punkte)

1. Kurze Fragen.

- a) (4 Punkte) Wir haben formale Modelle zur Beschreibung von asynchronen nebenläufigen Vorgängen kennengelernt. Erklären Sie kurz die Begriffe *formales Modell* und *nebenläufiger Vorgang*.
- b) (3 Punkte) Die Zahl der Marken in einem markierten Graphen bleibt immer konstant. Was können Sie über einen solchen Graphen aussagen?

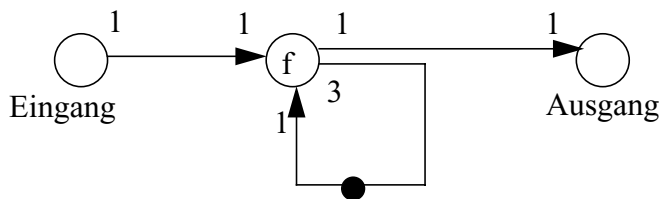
2. Markierte Graphen



Im oben abgebildeten markierten Graphen, stellt der mit + gekennzeichnete Knoten eine Addition der beiden Eingangsdaten des Knotens dar.

- a) (4 Punkte) Am Eingang a wird eine Zahlensequenz eingelesen wobei $a(k)$ den k -ten Wert bezeichnet. Geben Sie die Ausgangssequenz $b(k)$ als Funktion der Eingangswerte an.
- b) (6 Punkte) Die Anfangsmarkierung mit dem Wert s wird durch n Markierungen ersetzt. Geben Sie die Rekursionsformel für die Ausgangssequenz $b(k)$ an.

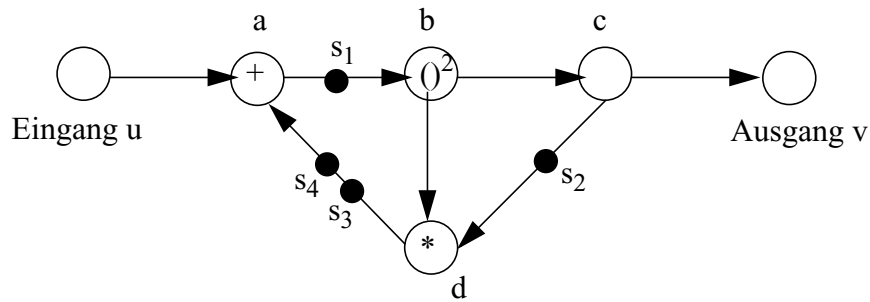
3. Synchrone Datenflussgraphen



Der oben abgebildete synchrone Datenflussgraph ist aus dem bereits verwendeten markierten Graphen abgeleitet. Der Knoten führt die Funktion f auf den Eingangsdaten aus.

- a) (4 Punkte) Welche Probleme könnten entstehen, wenn dieser Graph in ein Softwareprogramm umgesetzt wird?
- b) (2 Punkte) Ändern sie die Raten und die Anfangsmarkierung *in der Schleife* so, dass der Ausgangsknoten höchstens 3 Mal feuern kann. Eine Rate mit dem Wert 0 ist nicht zulässig.
- c) (4 Punkte) Setzen sie den Graphen aus Teilaufgabe (b) in ein Programm in der Sprache C um. Dabei können sie die Prozeduraufrufe `readInput()`, `writeOutput(...)`, `readQueue()` und `writeQueue(...)` verwenden, die ein Datum vom Eingang zurückliefern, ein Datum auf den Ausgang schreiben, ein Datum von einer Warteschlange lesen beziehungsweise ein Datum in eine Warteschlange schreiben. Die Lese-Prozeduren warten jeweils, bis die entsprechenden Daten vorliegen. Am Anfang ist die Warteschlange leer.

4. In der nächsten Teilaufgabe geht es um den folgenden markierten Graphen:



s_1 bis s_4 sind die Datenwerte der Anfangsmarkierungen. Die Knoten a , b , c und d führen die Funktionen *Addition*, *Quadrieren*, *Weiterleiten* und *Multiplizieren* auf den jeweiligen Eingangsdaten aus. Im Fall mehrerer Ausgangskanten haben die bei einer Feuerung generierten Daten die gleichen Werte.

a) (8 Punkte) Am Eingang u wird eine Zahlensequenz $u(k)$ eingelesen. Geben Sie die Ausgangssequenz $v(k)$ als Funktion der Eingangswerte an.