

Lagerhaltung

Dickbauer Y., Moser P., Perner M.

PS Computergestützte Modellierung, WS 2016/17

December 19, 2016

Outline

- 1 Aufgabenstellung
- 2 Flow Chart
- 3 Programmcode
 - Main Funktion
 - Verwendete Funktionen
- 4 Beispiel

Aufgabenstellung

Ein Großhändler steht vor folgendem Lagerhaltungsproblem: Die Lagerung eines Produktes kostet 2 Euro pro Tag, eine Bestellung beim Lieferanten verursacht Kosten von 20 Euro (unabhängig von der Bestellmenge), der Verdienstentgang für ein nachgefragtes, aber nicht vorhandenes Produkt beträgt 15 Euro.

Die tägliche Nachfragemenge sei

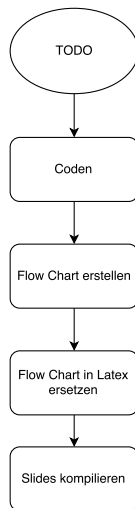
- ① binomialverteilt mit $n = 7$ und $p = 0.5$
- ② poissonverteilt mit $\lambda = 3$.

Aufgabenstellung

Die Bestellregel für den Disponenten soll lauten: Bestelle q Stück, wenn weniger als s Stück auf Lager sind. Bestimmen Sie dazu durch Simulation kostenoptimale Werte q , s (Variation von q und s zwischen 2 und 8 Stück mit $q \geq s$).

- Eingabe: -
- Output: Lagerbestand zu Periodenstart, Einkauf, Nachfrage, Verdienstentgang und Gesamtkosten je Periode

Flow Chart



Main Funktion - Programmeinstieg

```
1 def main():  
2     pass
```



Funktion `user_input(input_vars, [use_defaults])`

- Diese Funktion verlangt vom User die geforderten Eingabeparameter und gibt diese als von der Programmiererin gewünschten Datentyp wieder zurück
- Funktion verlangt als ersten Eingabeparameter die Liste *input_vars*
- Falls *use_defaults == True* wird der User nicht nach Eingabe gefragt (Dient zum Testen)
- Diese Liste besteht wiederum aus Listen mit je Länge = 3:
 - 0: Text, welcher dem User ausgegeben wird
 - 1: Datentyp (int/float/str)
 - 2: Default value: Dieser Wert wird zurueckgegeben, falls *use_defaults == True*

```
1 x, y = user_input((
2     ('Geben_Sie_einen_X_Wert_ein', int, 10),
3     ('Geben_Sie_einen_Y_Wert_ein', int, 5), False):
```



Beispiel anhand fixer Zufallszahlen

- Annahme der Zufallszahlen wie folgt:

iteration	0	1	2	3
ZZ	1	2	3	4

blub