

Fertigungsstraße

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

Eine Fertigungsstraße besteht aus 3 Maschinen. Die Straße erhält die zu bearbeitenden Teile aus einem Rohlager (mit unendlichem Vorrat); die bearbeitenden Teile werden in ein nachgeschaltetes Fertigteilager (mit ebenfalls unendlicher Kapazität) geliefert.

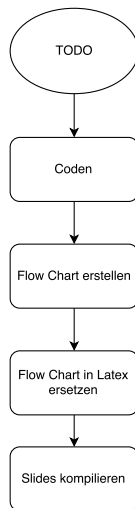
Die normale Bearbeitungszeit eines Teiles beträgt pro Maschine eine Minute; bei 15 Teilen tritt jedoch eine Störung von vier Minuten auf, so dass die gesamte Bearbeitungszeit eines Teiles dann 5 Minuten pro Maschine beträgt.

Aufgabenstellung

Untersuchen Sie durch Simulation über 180 Minuten, ob es zweckmässig ist, die drei Maschinen zu entkoppeln und zwischen den Maschinen ein Pufferlager einzurichten. Vergleichen Sie dazu beispielsweise den Ausstoß (gefertigte Stückzahl), die Stillstandszeiten und den Auslastungsgrad der Maschine sowie die durchschnittliche Bearbeitungszeit eines Teiles auf der Fertigungsstraße. Stellen Sie im Rahmen der Präsentation den Ablauf des Programmes anhand von selbstgewählten Zufallszahlen beziehungsweise mit einem Gantt-Diagramm die Abhängigkeiten der Maschinen vor.

- Eingabe: -
- Output: Verlauf von Produktion (Startzeit, Bearbeitungszeit, Endzeit je Produkt und Maschine), Warteschlangenlänge bei Bearbeitung und Inspektion, sowie die oben angeführten Kennzahlen. Kennzahlen.

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