

Modul Aktivitätsdiagramm und Algorithmen

Einführung

Ausgangsproblem:

Wir wollen prüfen, ob ein Element in einer Liste vorhanden ist: 1,3,5,8,9,10,21,22,27,29, ...
Wie könnte ein Computer dieses Problem lösen? Natürlich mit einem Algorithmus. Doch was ist überhaupt ein Algorithmus?

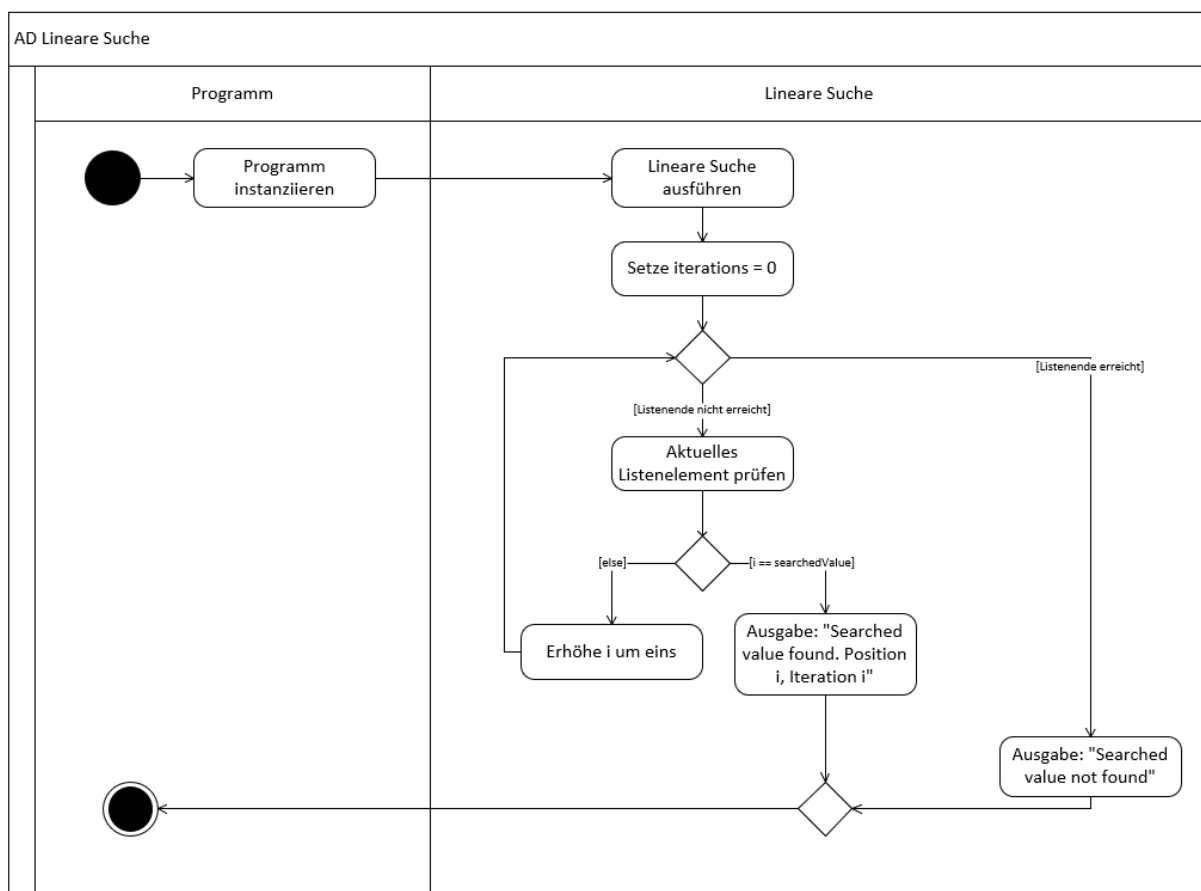


„Ein **Algorithmus** ist eine eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen. Algorithmen bestehen aus endlich vielen, wohldefinierten Einzelschritten. Damit können sie zur Ausführung in ein Computerprogramm implementiert, aber auch in menschlicher Sprache formuliert werden.“

(Quelle: <https://de.wikipedia.org/wiki/Algorithmus>)

Lineare Suche

Die lineare Suche ist ein Algorithmus, der versucht ein Element in einer Liste oder einem Array mit n Elementen zu finden, indem er die Liste Element für Element durchgeht, bis man es gefunden hat.



Lineare Suche als Pythoncode:



<https://t1p.de/sd1lq>



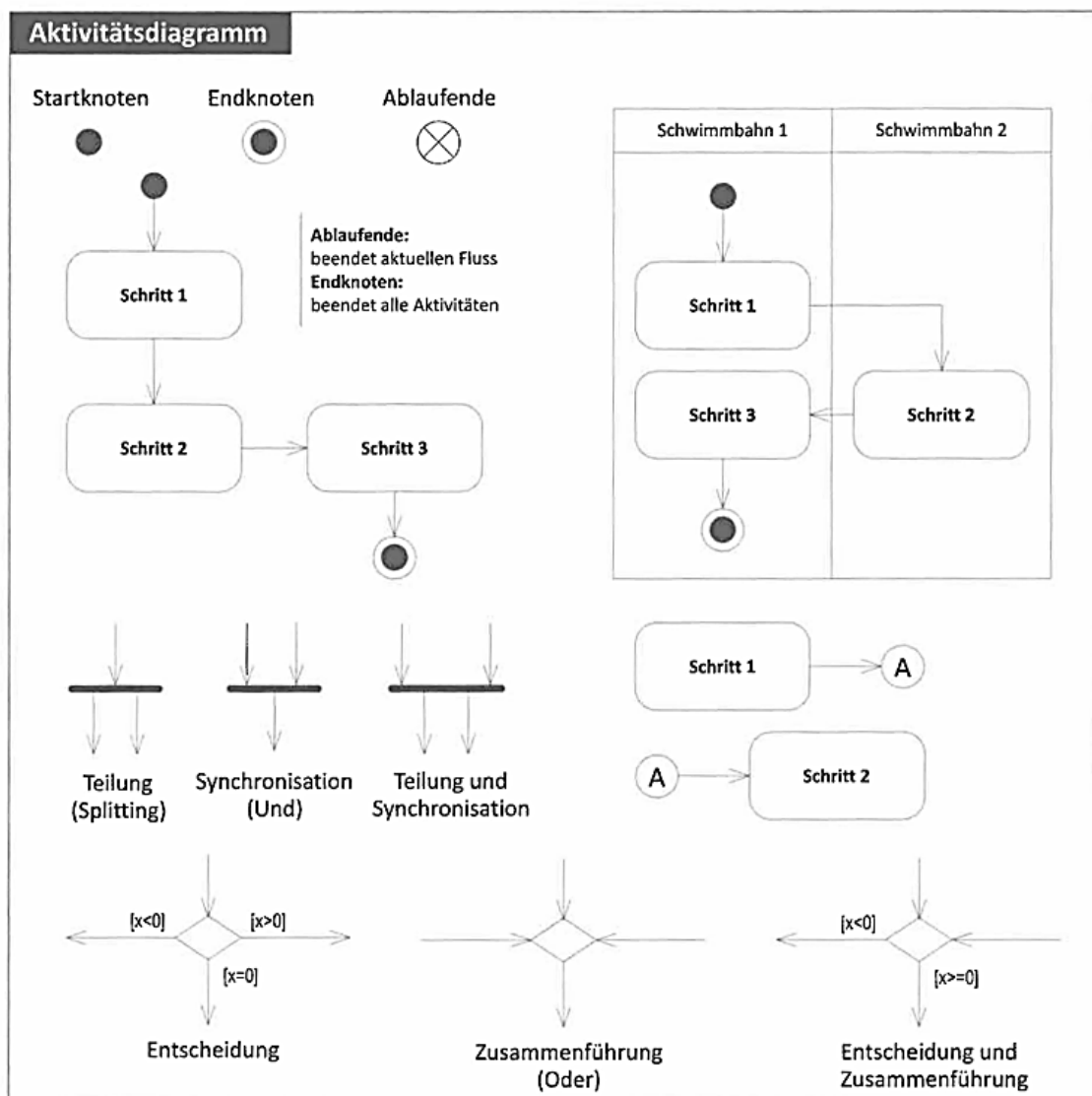


Einführung Aktivitätsdiagramm

- beschreibt aus der Verhaltenssicht den Ablauf eines Geschäftsfalls. Es setzt damit die noch recht abstrakten Vorgaben des Anwendungsfalldiagramms in konkrete Abläufe um. Diese Abläufe berücksichtigen einerseits den Standardablauf als auch mögliche Ausnahmen und Verzweigungsmöglichkeiten.
- Ziel eines solchen Diagramms wird es in den seltensten Fällen sein, das Verhalten des kompletten Systems zu zeigen, sondern vielmehr einen kritischen Teilausschnitt des Systems zu visualisieren.

Cheat-Sheet(Belegsatz) der IHK:

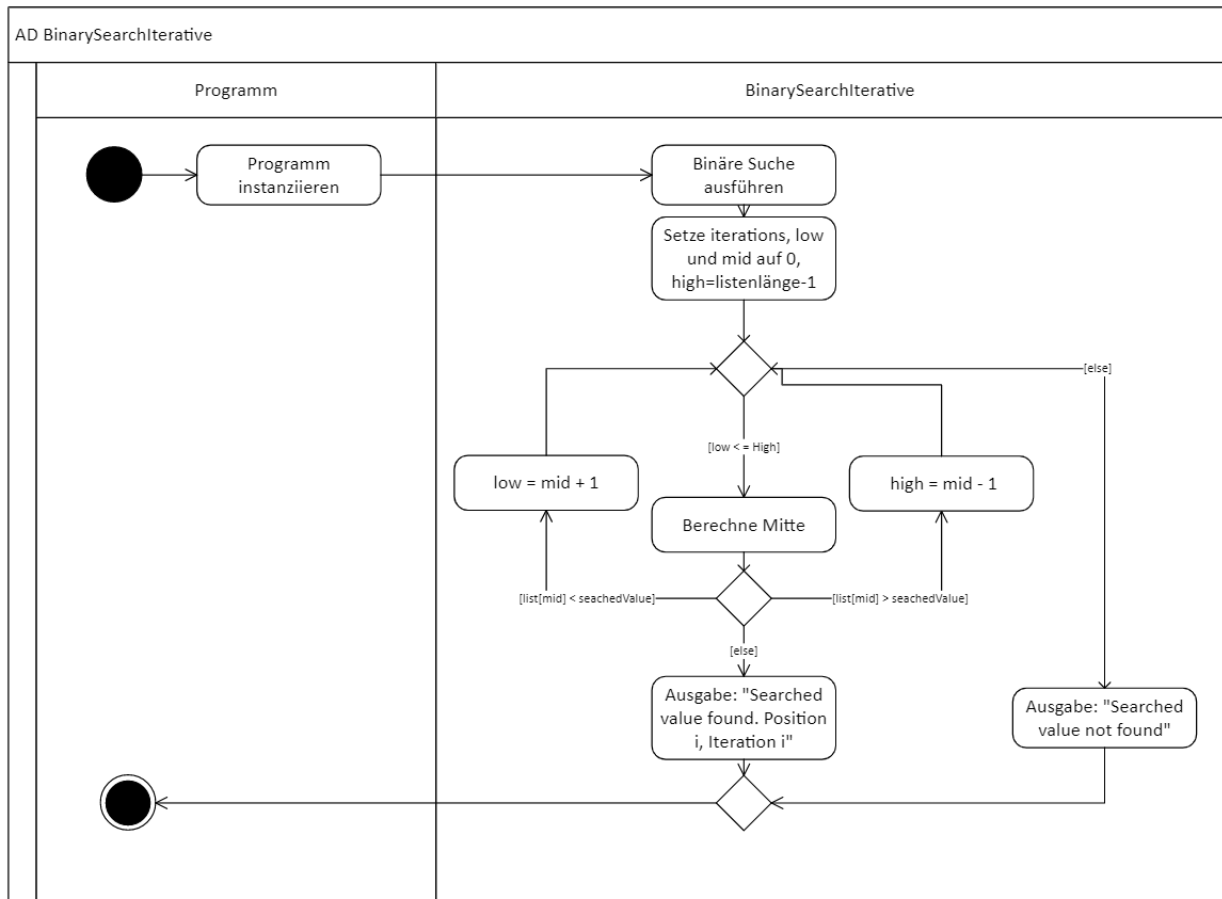
UML-Aktivitätsdiagramm



Hinweis: Das Symbol Ablaufende kann z.B. dann sinnvoll eingesetzt werden, wenn nach einer Teilung (Parallelisierung) einer der aktuellen Flüsse beendet wird, der andere jedoch weiter fließt.

Binäre Suche

Die binäre Suche ist ebenfalls ein Suchalgorithmus, der auf in einer Liste effizient ein gesuchtes Element finden kann. Der Algorithmus funktioniert nach dem Prinzip „Teile und Herrsche“. Hierzu wird zuerst das mittlere Element der Liste geprüft und die anschließend die vordere Hälfte oder die hintere Hälfte der Liste wiederum in der Mitte geprüft. Die jeweils andere Hälfte wird nicht weiter betrachtet.



Binäre Suche als Pythoncode:

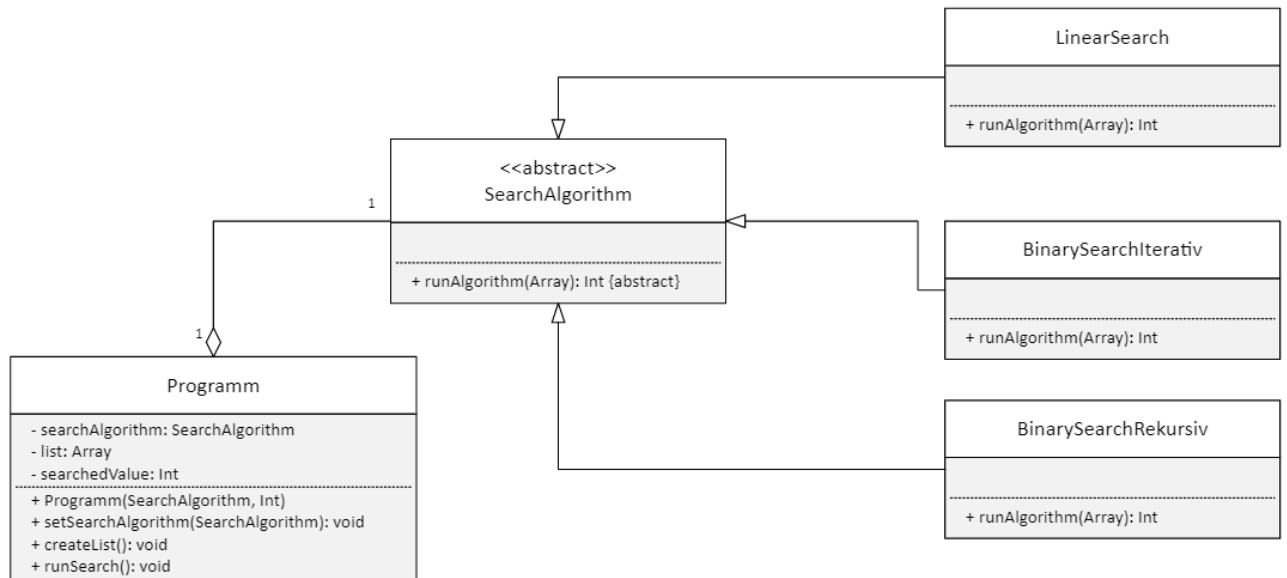


<https://t1p.de/c7bq0>



Programmstruktur und Erweiterung

Einfache Programmstruktur



"Modules should be both open (for extension) and closed (for modification)."

„Module sollten sowohl offen (für Erweiterungen) als auch verschlossen (für Modifikationen) sein.“

– BERTRAND MEYER: Object Oriented Software Construction^[1]

Open-Closed Prinzip

Quelle: <https://de.wikipedia.org/wiki/Open-Closed-Prinzip>



Aufgabe: Wie könnte ein Klassendiagramm aussehen, das einen Programm dynamisch zur Laufzeit auch ermöglichen würde, den Sortieralgorithmus auszutauschen?

Lösungsvorschlag Klassendiagramm

	https://t1p.de/iyp0o	
--	---	--

Lösungsvorschlag Programmcode

	https://t1p.de/bes5o	
--	---	--

Übungen



Bankautomat (vereinfacht)

Für einen Bankautomat soll der Vorgang „Geld abheben“ anhand eines Aktivitätsdiagramms dargestellt werden. Der Vorgang „Geld abheben“ wird wie folgt beschrieben:

- Der Kunde gibt seine EC-Karte ein.
- Der Geldautomat überprüft die EC-Karte. Wenn die EC-Karte nicht gültig ist, wird sie einbehalten und der Vorgang abgebrochen.
- Wenn die EC-Karte gültig ist, muss der Kunde seine PIN-Nummer eingeben.
- Der Geldautomat überprüft die PIN-Nummer. Wenn die PIN-Nummer nicht gültig ist, kann der Benutzer die PIN-Nummer bis zu drei Mal eingeben. Sofern die PIN-Nummer auch nach der dritten Eingabe falsch ist, wird die Karte einbehalten und der Vorgang anschließend abgebrochen.
- Sofern die eingegebene Nummer gültig ist, gibt der Kunde den gewünschten Betrag ein.
- Der Geldautomat aktualisiert das Konto und gibt anschließend die Karte wieder aus. Anschließend gibt der Geldautomat das Geld aus. Der Vorgang ist beendet.

Erstellen Sie unter Einbezug von passenden Verantwortungsbereichen ein Aktivitätsdiagramm.

Dreiwege-Handshake in TCP

Der Client, der eine Verbindung aufbauen will, sendet dem Server ein SYN-Paket. Der Server prüft den Ziel-Port hinsichtlich seiner Verfügbarkeit. Ist der im SYN-Paket angegebene Ziel-Port geschlossen, antwortet der Server mit einem TCP-RST, um zu signalisieren, dass keine Verbindung aufgebaut werden kann. Ist der Port geöffnet, bestätigt er den Erhalt des ersten SYN-Pakets und stimmt somit auch implizit dem Verbindungsaufbau zu. Der Client bestätigt zuletzt den Erhalt des SYN/ACK-Pakets durch das Senden eines eigenen ACK-Pakets.

Online-Terminvergabe

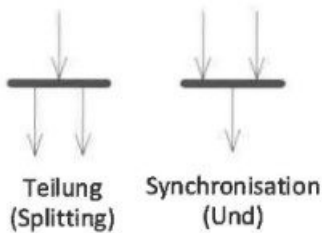
Sie wollen für eine Arztpraxis die Terminvergabe optimieren und hierbei einen Online-Anmeldung ermöglichen. Dabei soll folgender Ablauf realisiert werden.

Patient ruft die Terminreservierung auf, wonach er eine Anzeige der Ärzte und Terminarten erhält. Jetzt kann er den Arzt bzw. die Terminart auswählen. Nach der Auswahl sucht die Software den 1. freien Termin. ...	
Variante 1:	Variante 2:
... Das Programm zeigt den Monat, in dem der 1. freie Termin liegt, mit allen freien Terminen im angezeigten Zeitraum an. Der Anwender wählt einen freien Termin. Das System versucht nun, den Termin zu blockieren. Wenn der Termin noch frei ist, kann er final reserviert und parallel dazu der Anwender informiert werden. Danach ist die Aktivität beendet. Ist der Termin inzwischen nicht mehr verfügbar, wird eine entsprechende Meldung ausgegeben und erneut der 1. freie Termin gesucht, um dem Anwender die Auswahl anzuzeigen.	... Die Software ermittelt den Monat des 1. freien Termins. Es werden alle Termine des ermittelten Monats blockiert. Der Monat mit allen freien Terminen wird zur Auswahl angezeigt. Der Anwender kann einen freien Termin wählen. Danach wird der Anwender bzgl. der erfolgreichen Reservierung informiert und parallel der Termin final reserviert sowie alle anderen blockierten Termine freigegeben. Danach ist die Aktivität beendet.

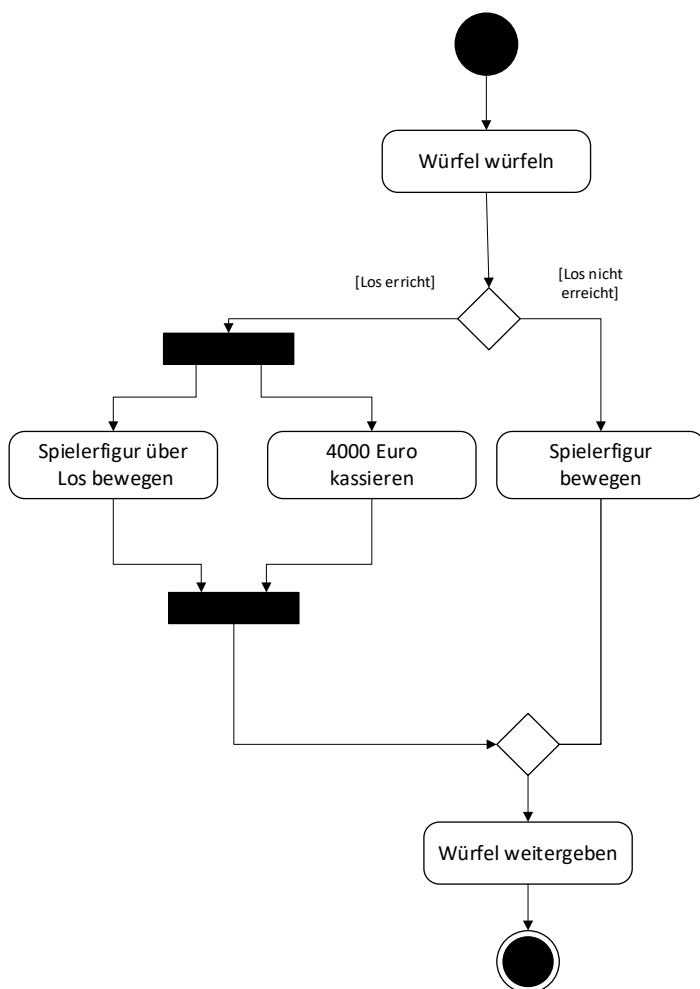
Erstellen Sie zu der Variante 1 ein passendes Aktivitätsdiagramm inklusive Verantwortungsbereiche unter Beachtung des nachfolgenden Hinweises.

Wichtiger Hinweis:

Sollte ein Prozess sich in zwei oder mehr parallellaufende Prozesse aufteilen, ist zuerst das Teilungssymbol und später bei der Wiedervereinigung der Prozesse das Synchronisationssymbol zu notieren.



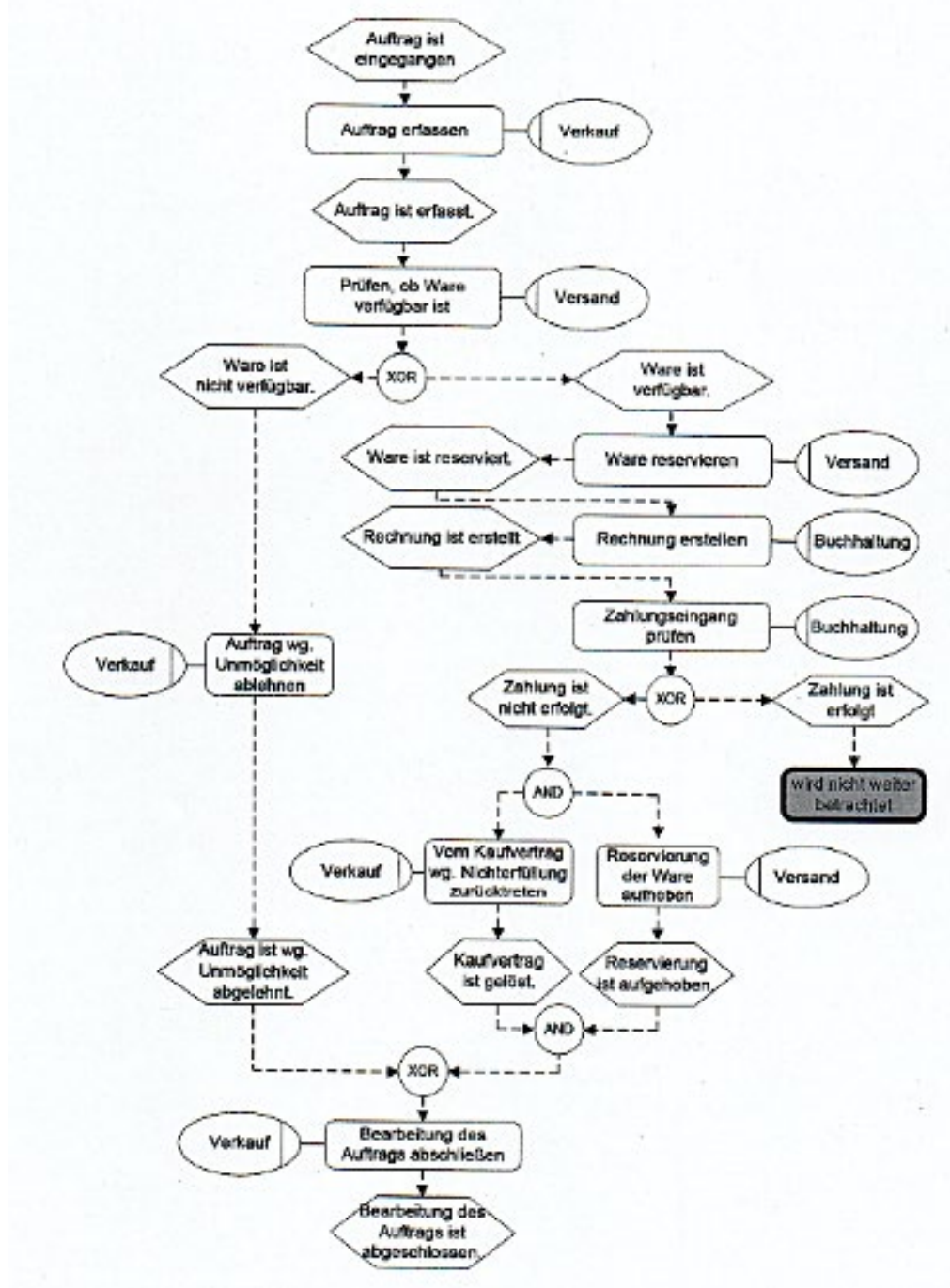
Beispiel: Synchronisation von nebenläufigen Aktivitäten anhand des Losüberschreitens im Spiel Monopoly.



Wie obiges Diagramm darstellt, ist es auch möglich, Aktionen parallel durchzuführen (hier Spielerfigur über Los zu bewegen und 4000 Euro zur gleichen Zeit zu kassieren).

EPK-Überführung

Ein Kaufmann für Digitalisierungsmanagement hat ein EPK für den Geschäftsprozess „Auftragsabwicklung“ erstellt. Diesen sollen Sie in ein Aktivitätsdiagramm überführen.



Kurierfahrt

Während einer Kurierfahrt werden folgende Tätigkeiten ausgeführt:

- Auftraggeber ruft bei der Zentrale an und gibt den Auftrag auf.
- Die Zentrale erfasst die Auftragsdaten des Auftraggebers.
- Die Zentrale leitet den Auftrag an den Kurier weiter.

- Der Kurier erfasst bei Abholung beim Auftraggeber die genauen Auftragsdaten und übermittelt diese anschließend an die Zentrale.
- Der Kurier führt die Fahrt durch.
- Bereits während der Kurier die Fahrt durchführt, gleicht die Zentrale die vom Kurier gesendeten Auftragsdaten mit den bei Auftragserteilung erfassten Daten ab.
- Der Kurier liefert die Sendung ab.
- Danach meldet der Kurier den Abschluss der Fahrt.
- Die Zentrale erstellt, nachdem der Abschluss der Fahrt gemeldet wurde sowie nach obig genannter Datenüberprüfung eine Rechnung.