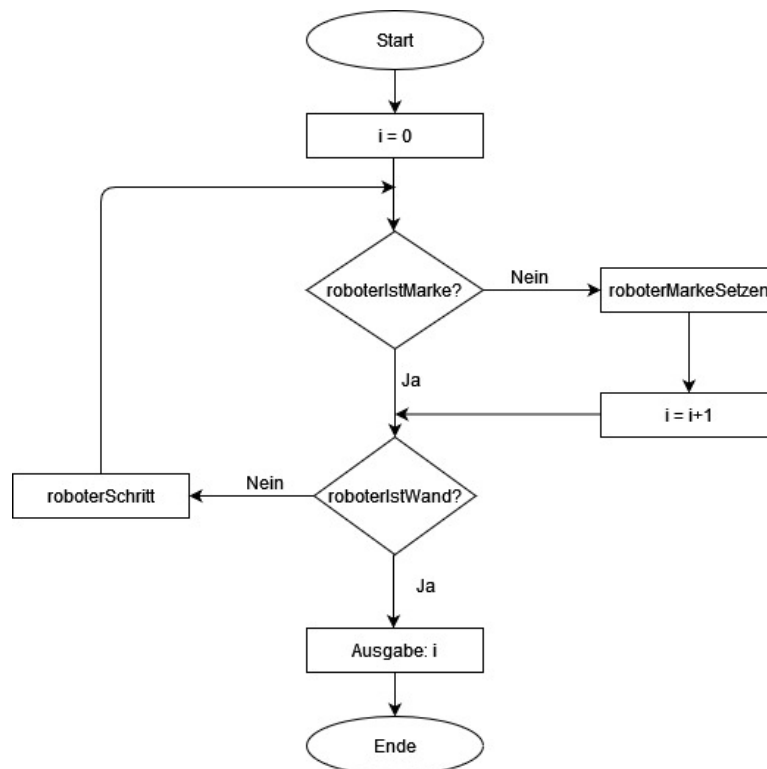


8.2 Visualisierung algorithmischer Abläufe: Programmablaufplan, Struktogramm, Aktivitätsdiagramm

In der Praxis sind Algorithmen häufig sehr komplex. Um diese besser nachzuvollziehen, haben sich verschiedene Visualisierungsformen entwickelt. Die ersten waren dabei Programmablaufpläne und etwas später Struktogramme.

8.2.1 Programmablaufplan (DIN 66001)

Ein Programmablaufplan könnte wie folgt aussehen:

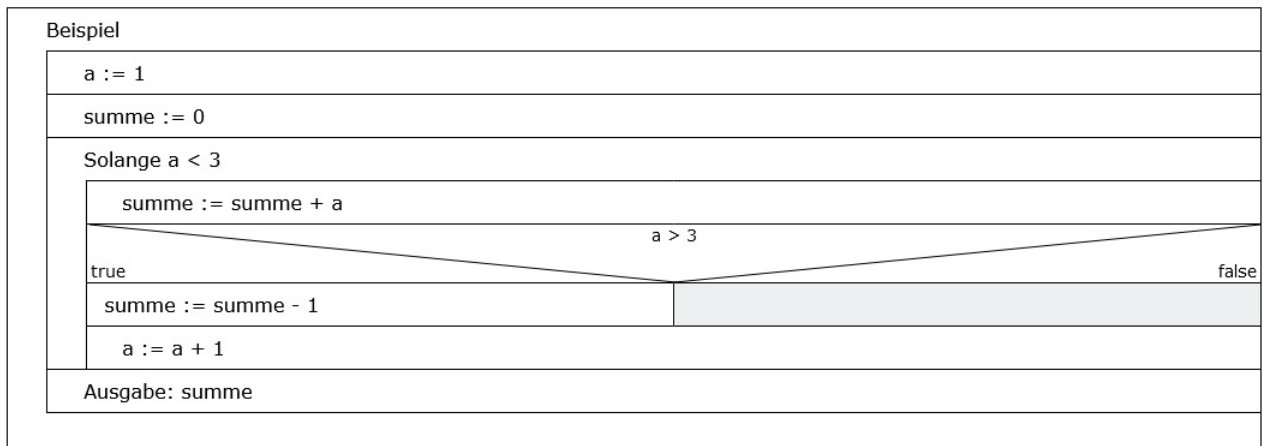


Aufgaben:

1. Informieren Sie sich mithilfe von diesem Beispiel über die Bedeutung der verschiedenen Symbole in einem Programmablaufplan.
2. Welche Kontrollstrukturen treten im obigen Beispiel auf? Beschreiben Sie kurz die Funktion von diesem Programm.
 - Loops
 - If / Else
 - Anweisungen

8.2.2 Struktogramm (DIN 66261)

Eine andere Visualisierungsmöglichkeit bietet ein Struktogramm bzw. Nassi-Shneiderman-Diagramm. Ein solches Diagramm könnte zum Beispiel wie folgt aussehen:



Aufgaben:

1. Informieren Sie sich über die jeweilige Darstellung von Anweisungen und der verschiedenen Kontrollstrukturen (Verzweigung, kopfgesteuerte und fußgesteuerte Schleife) in einem Struktogramm bzw. Nassi-Shneiderman-Diagramm. Beschreiben Sie kurz das Beispiel-Struktogramm.

Addiert die Zahlen von 1 bis 2

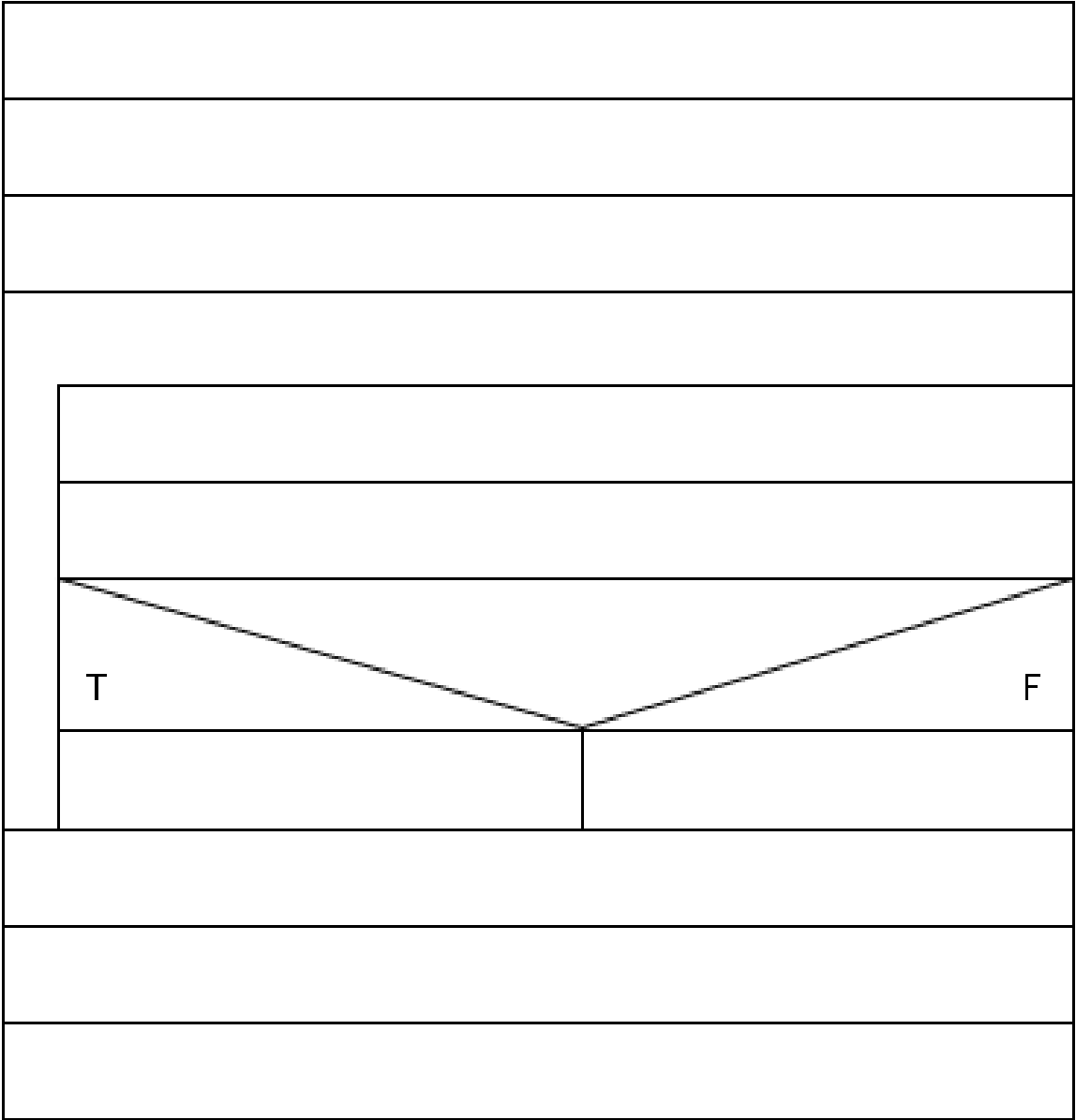
2. Setzen Sie den Programmablaufplan aus dem vorherigen Abschnitt als Struktogramm um.

Setzen Sie dieses Struktogramm in einen Programmablaufplan um

3. Die Daten einer Bestellung wurden bisher in einer Datei „Bestellungen.csv“ gespeichert. Diese sollen nun durch ein Programm ausgelesen und in eine Datenbank „Verkaufsdaten“ übertragen werden. Weiterhin sollen alle Bestellungen ab dem Jahr 2021 gezählt und am Ende ausgegeben werden. Zusätzlich sollen geöffnete Verbindungen wieder geschlossen werden.

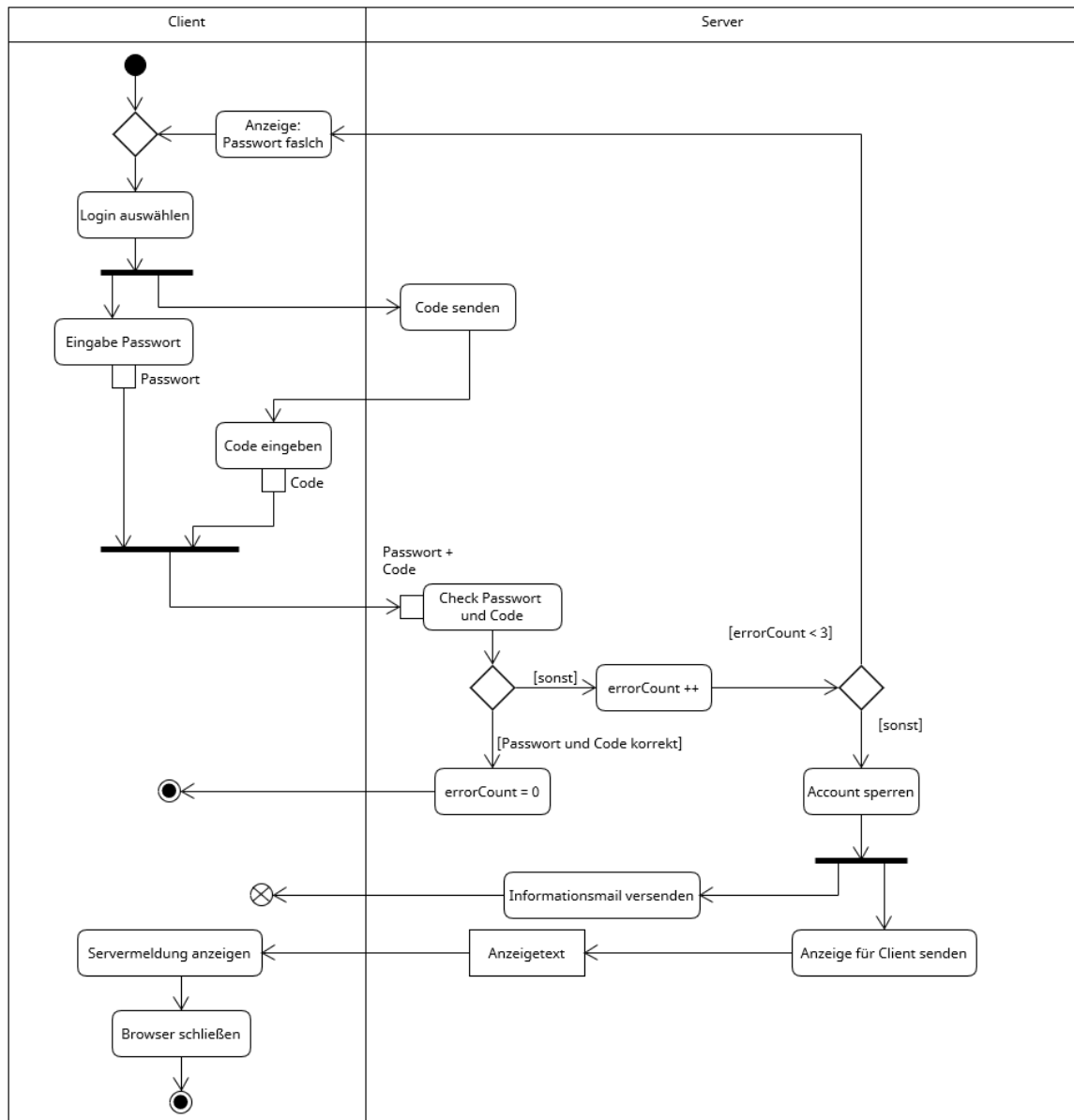
Vervollständigen Sie für diese Aufgabe das auf der nächsten Seite abgebildete Struktogramm, hierfür stehen folgende Hilfsmethoden zur Verfügung:

- `openDatabase(String databaseName):`
Öffnet die entsprechende Datenbank und liefert eine Referenz auf die Datenbank zurück.
- `openFile(String file):`
Öffnet die entsprechende Datei und liefert eine Referenz auf diese Datei zurück.
- `<<Referenz>>.close():`
Schließt die entsprechende Referenz (Datenbank oder Datei).
- `<<Referenz>>.readDataRecord():`
Liest aus einer geöffneten CSV-Datei (Referenz) einen kompletten Datensatz einer Bestellung und gibt diesen als String zurück. Anschließend wird zum nächsten Datensatz in der Datei gegangen, falls noch nicht das Ende der Datei erreicht worden ist.
- `<<Referenz>>.writeDataRecord(String dataRecord):`
Schreibt in eine geöffnete Datenbank (Referenz) die Daten einer Bestellung, welche der Funktion als Datensatz übergeben wird.
- `getYear(String dataRecord):`
Ermittelt aus dem übergebenen Datensatz das Bestelljahr und gibt diesen zurück.
- `<<Referenz>>.endOfFile():`
Gibt zurück, ob das Ende einer Datei (Referenz) erreicht wurde.



8.2.3 UML Aktivitätsdiagramm

Programmablaufpläne und Struktogramme werden heutzutage in der Softwareentwicklung kaum noch verwendet. Diese wurden größtenteils durch UML-Aktivitätsdiagramme abgelöst. Ein solches Diagramm könnte dabei wie folgt aussehen:



Aufgaben:

1. Informieren Sie sich über die Bedeutung der verschiedenen Symbole in einem UML-Aktivitätsdiagramm. Erklären Sie kurz in eigenen Worten obiges Aktivitätsdiagramm.

2. Modellieren Sie folgenden Prozess mittels eines UML-Aktivitätsdiagramms:
- Am Schuljahresende ist in Ihrer Klasse ein Weißwurstfrühstück angesagt: Zunächst werden in Ihrer Klasse die Aufgaben verteilt. Um noch vor 12 Uhr die Weißwürste vertilgen zu können, werden 4 Gruppen gebildet, die gleichzeitig loslegen. Gruppe 1 kümmert sich in der Küche um das Erwärmen des Wassers. Gruppe 2 besorgt beim Metzger Weißwürste, während Gruppe 3 sich um die Brezen kümmert. Wenn das Wasser heiß ist und die Weißwürste besorgt wurden, kann mit dem Erhitzen der Weißwürste begonnen werden.
- Gruppe 4 ist für alle weiteren Besorgungen zuständig (Senf, Weißbier etc.). Sie geht dabei folgendermaßen vor: Zunächst sucht sie einen Supermarkt und kauft schließlich dort ein. Hat sie alle Zutaten bekommen, kehrt sie zur Schule zurück, ansonsten sucht sie solange einen weiteren Supermarkt und kauft dort ein bis alle Besorgungen erledigt sind. Nachdem alle Gruppen mit Ihren Aufgaben fertig sind, beginnt das Weißwurstfrühstück.

3. Für einen Bestellvorgang eines Onlineshops wurden folgende Aktionen (ohne Festlegung der Reihenfolge) identifiziert:

Kunde	Onlineshop
Einloggen	Ware versenden
Zur Kasse gehen	Ware packen
Adresse eingeben	Bestellbestätigung schicken
Zahlungsart eingeben	Rechnung erstellen
Artikel ansehen	
Artikel in den Warenkorb legen	
Als Gast bestellen	
Kaufauftrag erteilen	

Erstellen Sie für die folgende Abfolge der Aktionen ein UML-Aktivitätsdiagramm. Nutzen Sie dafür die Vorlage auf der nächsten Seite.

- Der Kunde kann beliebig viele Artikel ansehen und ausgesuchte Artikel in den Warenkorb legen.
- Die Auswahl kann beendet werden oder ist abgeschlossen, wenn der Kunde zur Kasse geht.
- Ein bereits registrierter Kunde kann sich mittels Kunden-Login anmelden, alternativ ist eine Bestellung als Gast möglich.
- Ein Gast muss seine (Liefer-)Adresse angeben.
- Sowohl ein Gast als auch ein Kunde müssen die Zahlungsart auswählen.
- Nach der Wahl der Zahlungsart kann der Kundenauftrag erteilt werden.
- Der Onlineshop bestätigt die Bestellung.
- Der Onlineshop erstellt die Rechnung und verpackt gleichzeitig die Ware.
- Die Ware wird versendet.
