

Programmierung A/B (in Java) – Übung 7

Thema: Bäume und Rekursion

Lernziele: Das Verständnis von und der praktische Umgang mit Baumstrukturen, sowie die Verwendung von Rekursion am Beispiel einer Baumsuche.

Aufgabe 1: Bäume

In einem Königreich der Mittel Erde haben alle Mitglieder der Königsfamilie entweder 1 oder 2 Kinder. Das Königreich wird von einem König/einer Königin regiert. Ein Mitglied der Königsfamilie darf nur den Thron besteigen, wenn er/sie mindestens 16 Jahre alt ist.

Erstellen Sie die Klasse *Person*, welche ein Mitglied der Königsfamilie darstellt. Jede *Person* hat einen Namen, einen Titel, ein Alter (in Jahren) und ein *Geschlecht*. Das *Geschlecht* soll durch einen Aufzählungstyp (*enum*) dargestellt werden mit den Werten MAENNLICH oder WEIBLICH. Jede Klasse vom Typ *Person* hat zudem zwei weitere Variablen, *erstesKind* und *zweitesKind*, die Referenzen zu den (maximal) zwei Kindern halten. Alle Variablen der Klasse sollen *private* Sichtbarkeit mit öffentlichen *getter* und *setter* Methoden haben.

Erstellen Sie auch eine Klasse *Koenigreich*, welche den Namen des Königreiches und eine Referenz auf den/die König/Königin speichert, sowie einen Aufzählungstyp *NachfolgerOrdnung* mit den Werten KINDER und GESCHWISTER definiert. Implementieren Sie weiterhin folgende Methoden der Klasse *Koenigreich*:

List<Person> thronNachfolgerNachKindern(Person person): Diese private Methode erstellt eine Liste aller Thronnachfolger der übergebenen Person. Die zurückgegebene Liste soll so geordnet sein, dass die Kinder der jüngsten Generation, ausgehend vom jeweils Erstgeborenen, zuerst ausgegeben werden. Anschließend folgen die Zweitgeborenen und zuletzt deren Eltern (Tiefensuche).

List<Person> thronNachfolgerNachGeschwistern(Person person): Diese private Methode erstellt eine Liste aller Thronnachfolger der übergebenen Person. Die Geschwister erben zuerst in Reihenfolge ihrer Geburt und dann kommen die Kinder, beginnend mit den Kindern der ersten Geschwister (Breitensuche).

void druckeThronNachfolger(NachfolgerOrdnung ordnung): Abhängig von der übergebenen Ordnung soll die Liste von Thronnachfolgern ausgegeben werden. Zu den Thronfolgern sollen der Name, der Titel und das Alter ausgegeben werden, zum Beispiel:

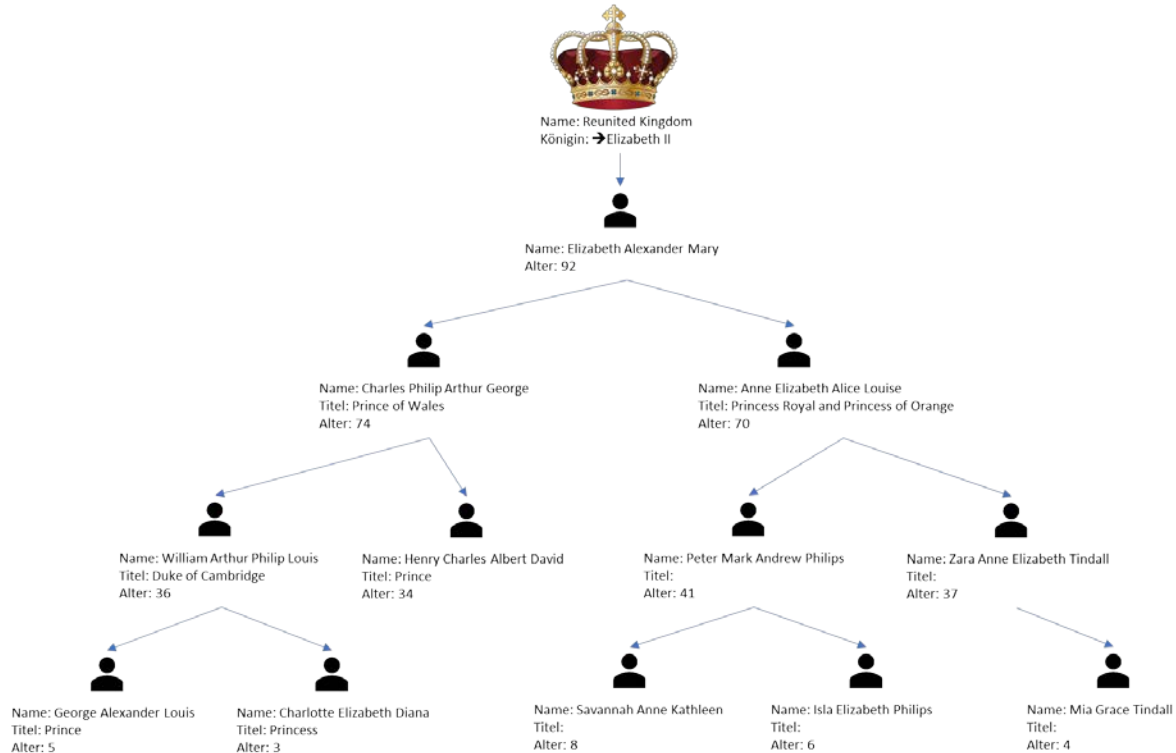
Mary Poppins, 36, Queen of Hearts

Hinweis: Beachten Sie bei der Implementierung das Mindestalter eines Thronnachfolgers!

Zusatzaufgabe: Versuchen sie, beide Methoden zur Bestimmung der Thronnachfolger sowohl iterativ als auch rekursiv zu implementieren.

Aufgabe 2:

Implementieren Sie eine Klasse Main mit einer main() Methode, die folgenden Familienbaum nachbaut:



Geben Sie anschließend die Liste der Thronnachfolger aus, sowohl nach der Ordnung KINDER, als auch der Ordnung GESCHWISTER.

Die korrekte Ausgabe der Methode „**thronNachfolgerNachKindern**“ wäre in diesem Fall:

William Arthur Philip Louis,	36,	Duke of Cambridge
Henry Charles Albert David,	34,	Prince
Charles Philip Arthur George,	74,	Prince of Wales
Peter Mark Andrew Philips,	41,	
Zara Anne Elizabeth Tindall,	37,	
Anne Elizabeth Alice Louise,	70,	Princess Royal and Princess of Orange

Die korrekte Ausgabe der Methode „**thronNachfolgerNachGeschwistern**“ wäre in diesem Fall:

Charles Philip Arthur George,	74,	Prince of Wales
Anne Elizabeth Alice Louise,	70,	Princess Royal and Princess of Orange
William Arthur Philip Louis,	36,	Duke of Cambridge
Henry Charles Albert David,	34,	Prince
Peter Mark Andrew Philips,	41,	
Zara Anne Elizabeth Tindall,	37,	