

Aufgabe 4 – Übungsaufgabe

- 1) Macht euch klar, dass die Datenstruktur Person rekursiv ist, denn jedes Objekt vom Typ Person referenziert zwei weitere Objekte vom Typ Person, nämlich Mom und Dad.
 - ➔ Die Datenstruktur Person ist rekursiv, jedes Objekt vom Typ Person referenziert seine/ihre Mutter und Vater, zwei weitere Objekte vom Typ Person. Das geschieht (in diesem Fall) bis zu einer Rekursionstiefe von 3, sodass 4 Generationen vorhanden sind.
- 2) Die Methode BuildTree() baut einen Beispiel-Baum auf. Setzt einen Breakpoint in [Zeile 19 von Program.cs](#), startet den Debugger und seht Euch den Inhalt von root im Debugger an.
 - ➔ root ist die Person "Willi Cambridge", Mom und Dad sind auch jeweils Personen, die wiederum auch ihre Mutter und Vater referenzieren.

- 3) Ändert die Bedingung so, dass nicht gleich die erste Person ("Willi") zurückgegeben wird. Eventuell gibt es Abstürze. Analysiert die Abstürze mit dem Debugger, überprüft Variableninhalte und den Call-Stack.

- ➔ Um nicht direkt "Willi" zurückzubekommen kann die Bedingung in Zeile 22 beispielsweise umgeändert werden zu:

```
if(person.LastName != "Cambridge")
```

Somit wird Willi's Mutter (Diana) zurückgegeben, da die Bedingung für Willi nicht wahr ist und im Stammbaum eine Generation nach oben gesprungen wird. Der Stammbaum wird zuerst auf mütterlicher Seite geprüft, deshalb Diana.

- 4) Schreibt komplexere Bedingungen, findet z.B. die erste Person, die in einer Altersspanne liegt, vergleicht dazu *person.DateOfBirth.Year* mit *DateTime.Now.Year*. Analysiert mit dem Debugger, ob Eure Bedingung richtig ist.

- ➔ Es lässt sich beispielsweise explizit nach einem bestimmten Namen suchen:

```
if(person.LastName == "York")
```

Zurückgegeben wird „Schorsch-Albert“.

Sucht man hier nach einem Namen, oder nach einer anderen Bedingung, die für keine Person im Stammbaum wahr ist, so wird eine *NullReferenceException* zurückgeworfen. Das passiert, weil die letzten Personen des Stammbaums keine Mutter bzw. keinen Vater mehr besitzen, diese Eigenschaften also „null“ sind.

In Zeile 22 wird mit *person.LastName* versucht auf die nicht existente Eigenschaft *LastName* von null zuzugreifen, was mit einer *NullReferenceException* fehlschlägt.