

Softwaredesign Aufgabe 2 – Debugging

Betrachtet die Source-Code-Dateien [FamilyTree.cs](#) und [Program.cs](#). In Familytree wird eine Datenstruktur aufgebaut, die eine Art Familienstammbaum repräsentiert (Ähnlichkeiten mit realen Personen sind rein zufällig :-)).

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 Funktion „public static Person Find (Person person)“ ruft sich nach dem durchlauf selbstständig wieder auf, bis die Funktion den Wert „null“ zurückgibt.

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.

Die Funktion „BuildTree()“ erzeugt einen Familienstammbaum der bei der jüngsten Person beginnt. Der Inhalt von „root“ im Debugger zählt die Familienmitglieder und ihre Abhängigkeit und umgekehrt chronologischer Reihenfolge auf. Beginnend mit dem jüngsten Familienmitglied Willi Cambridge.

Die Methode Find() durchläuft rekursiv den Baum und prüft alle Person-Objekte darauf, ob die Bedingung in [Zeile 22](#) gegeben ist. Die erste Person, die die Bedingung erfüllt, wird zurückgeliefert.

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.

Die if-Bedingung wird geändert:

Vorher: if (person.LastName != „Battenberg“)

Nachher: if (person.LastName != „Cambridge“ && person.LastName != „Spencer“)

Nun wird aufgrund der geänderten Suchparameter als erste Person Franzi Roche ausgegeben.

Wenn die Bedingung geändert wird auf:

```
If (person.LastName == „Battenberg“)
```

Entsteht eine „System.NullReferenceException“, hier wird nur die mütterliche Seite des Stammbaums durchsucht. Auf der mütterlichen Seite von Willis Familie ist jedoch niemand mit dem Nachnamen Battenberg. Diese „System-NullReferenceException“ kann mit einer kleinen Änderung am Code verhindert werden:

```
If(person.Mom != null)
```

```
    ret =Find(person.Mom);
```

```
If(person.Dad != null)
```

```
    ret =Find(person.Dad);
```

```
Return ret;
```

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.

```
Int age = DateTime.Now.Year – person.DateOfBirth.Year;
```

```
If(40 < age && age < 70)
```

```
Return person;
```

```
If(person.Mom != null)
```

```
    ret =Find(person.Mom);
```

```
If(person.Dad != null)
```

```
    ret =Find(person.Dad);
```

```
Return ret;
```

Dieser Code-Baustein ermöglicht es den Stammbaum nach Personen in einer bestimmten Altersspanne zu durchsuchen, hier zwischen 40 und 70 Jahren. Es wird dann die erste Person in dieser Altersspanne zurückgegeben.

