

Vorlesung Grundlagen der Programmiersprachen Sommer 2004 - Übungsblatt 5

Ausgabe: 06.07.2004 -- Abgabe: 14.07.2004, 13:00 Uhr, im Abgabekasten auf dem D3-Flur.

Aufgabe 13 (Laufzeitkeller, geschachtelte Prozeduren)

*** Aufgabe 13 und Aufgabe 15 werden bewertet. ***

Simulieren Sie für das folgende Programm den Laufzeitkeller bis zu dem Zeitpunkt, zu dem zum zweiten Mal der `print`-Befehl ausgeführt wird. Zeichnen Sie dazu eine Skizze nach dem Muster der rechts abgedruckten Tabelle. Kennzeichnen Sie entkellerte Schachteln durch Durchstreichen.

```

proc main()
{
  proc h(int a)
  {
    proc x(int b)
    {
      proc p()
      {
        int v = a;
        print v;
      }

      p();
    }

    proc y(int c)
    {
      x(c+1);
      if (c==1) h(3);
    }

    y(a)
  }

  h(1);
}

```

Prozedur-Name	Variablen	Statischer Vorgänger
main	---	
h	a:1	



Aufgabe 14 (Abstrakte Definition von Typen)

Da in Java keine Typen (außer Klassen) definiert werden können, müssen Klassen verwendet werden, um Daten zu strukturieren. Betrachten Sie das nachfolgende Java-Programm und beschreiben Sie die Typen, die der Programmierer im Sinn hatte, durch Wertemengen mit den Konzepten von Folie 505.

```
class Farbe {
    byte rot;
    byte gruen;
    byte blau;
}

abstract class Primitiv {
    // Dient als Oberklasse von Kreisen und Rechtecken
}

class Kreis extends Primitiv {
    Farbe farbe;
    Punkt mittelpunkt;
    int radius;
}

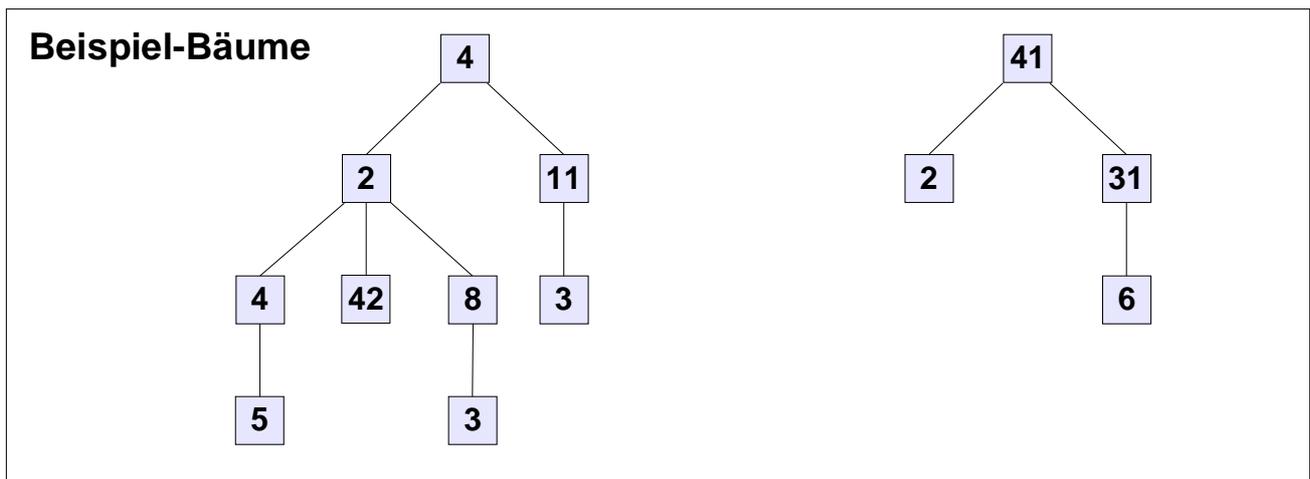
class Rechteck extends Primitiv {
    Farbe farbe;
    Punkt eckel, ecke2;
}

class Bild {
    Primitiv[] a = new Primitiv[100];
}
```

Aufgabe 15 (Rekursive Definition von Typen)

*** Aufgabe 13 und Aufgabe 15 werden bewertet. ***

Geben Sie eine abstrakte Typdefinition für Bäume an, in deren Knoten ganze Zahlen gespeichert sind. Jeder Baumknoten kann beliebig viele Kinder haben.



Zusatzaufgabe (wir nicht gewertet)

Entwickeln Sie Java-Klassendefinitionen für derartige Bäume aus Ihrer rekursiven Typdefinition. Geben Sie Anweisungen an, die den kleineren der abgebildeten Beispielbäume aufbauen.