

*Zentralübung zu  
Einführung in die Informatik I*

Dr. Christian Herzog  
Technische Universität München

Wintersemester 2000/2001

22. Januar 2001

# *Übersicht*

## **Klausur:**

- ❖ Wer darf teilnehmen?
- ❖ Wie sieht sie aus?

## **Programmieren mit Java:**

- ❖ Arrays, Referenz- vs. Wertsemantik
  - Aufgabe 39: Pascal-Dreieck
  - Aufgabe 40: Initialisierer

## *Teilnahmeberechtigt an der Klausur*

- ❖ Teilnahmeberechtigt sind
  - alle, die sich zu den Übungen eingeschrieben haben
  - und alle, die Informatik Diplom im 1. Semester studieren
  - und alle, die Informatik Bachelor im 1. Semester studieren
- ❖ Eine Liste aller Teilnahmeberechtigten hängt seit Dienstag, 16.1.01, im Schaukasten aus.
- ❖ Falls Sie nicht auf der Liste stehen, aber mitschreiben wollen/müssen:
  - email an [herzog@in.tum.de](mailto:herzog@in.tum.de)

## *Wie sieht die Klausur aus?*

- ❖ Wie sieht die Klausur aus?
  - Keine reinen Multiple-Choice-Aufgaben wie im WWW-Test
  - Die Aufgaben lehnen sich eher an die Art der Übungsaufgaben an.
  
- ❖ Wo finde ich alte Klausuren zum Üben?
  - die Vorlesung wurde von Prof. Brügge zum ersten Mal gelesen
  - es gibt deshalb keine alten Klausuren von ihm
  - Klausuren anderer Dozenten eignen sich nur bedingt zur Vorbereitung

## *Aufgabe 39 (Pascal-Dreieck): Die Klasse Funktionsdienst*

```
class Funktionsdienst {  
  
    Funktionsdienst() {}  
  
    public int[][] pascal(int n) {  
        int[][] dreieck = new int[n][];  
        for (int m=0; m<n; m++) {  
            dreieck[m] = new int[m+1];  
            dreieck[m][0] = 1;  
            dreieck[m][m] = 1;  
            for (int i=1; i<m; i++)  
                dreieck[m][i] = dreieck[m-1][i-1]  
                    + dreieck[m-1][i];  
        }  
        return dreieck;  
    }  
}
```

## *Aufgabe 39 (Pascal-Dreieck): Die Klasse Umgebung39*

```
class Umgebung39 {  
  
    public static void main(String[] args) {  
        Funktionsdienst f = new Funktionsdienst();  
        int zeilen = 5;  
        int[][] dreieck = f.pascal(zeilen);  
        for (int i=0; i<zeilen; i++)  
            for (int j=0; j<i+1; j++)  
                System.out.println(dreieck[i][j]);  
    }  
}
```

# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```

main

f:

zeilen:

dreieck:

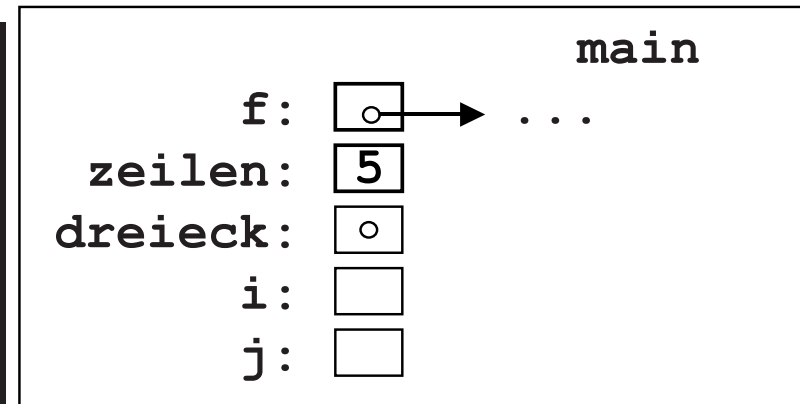
i:

j:

# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

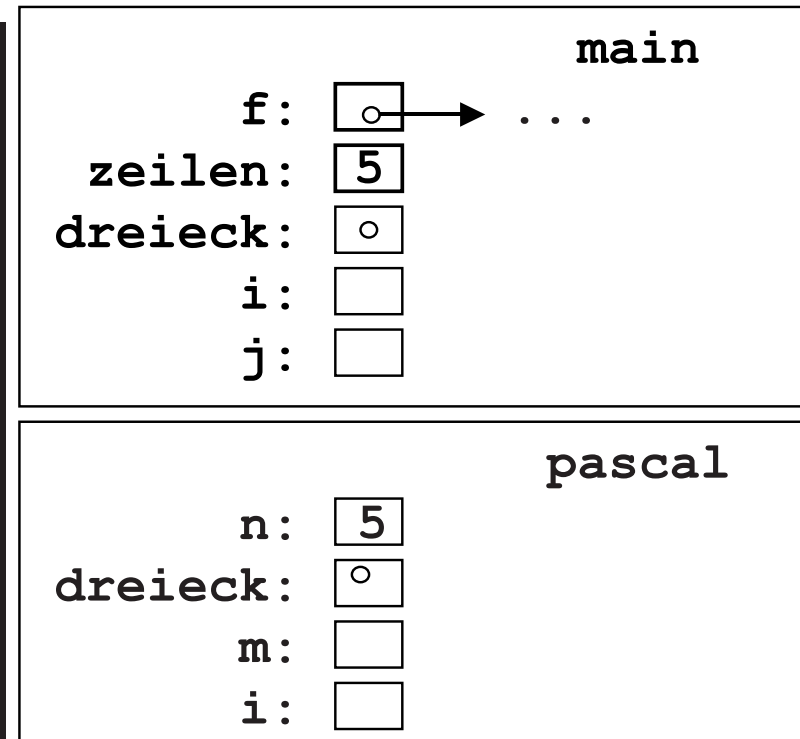
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

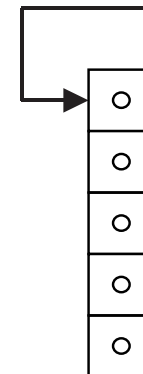
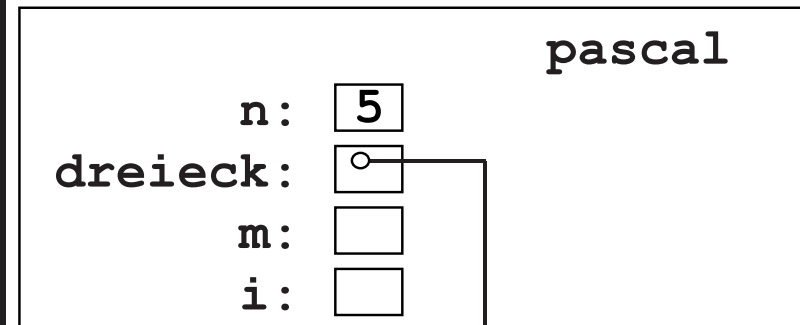
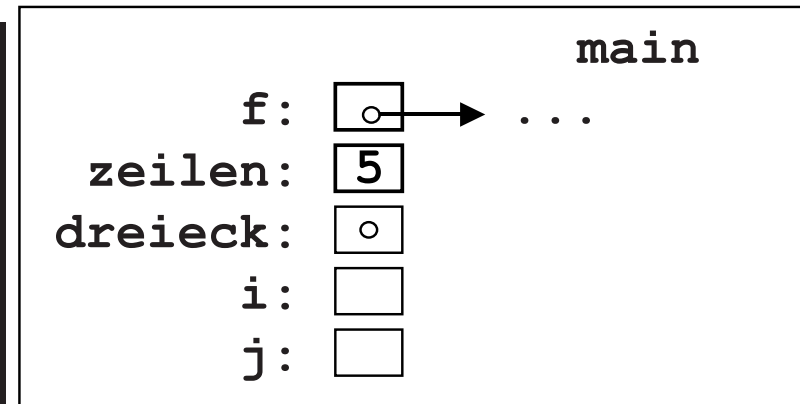
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

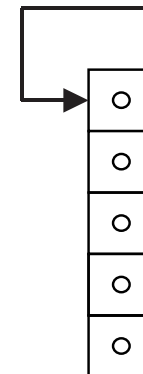
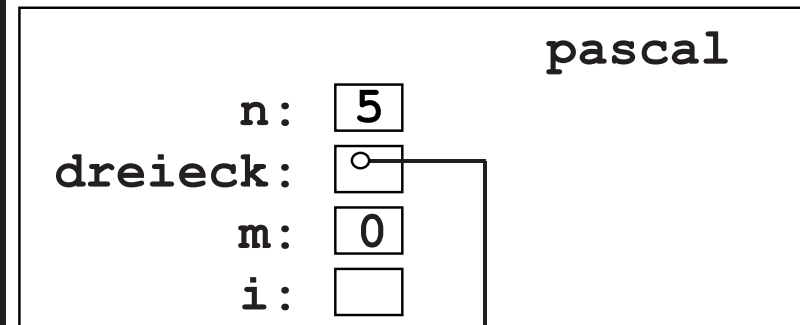
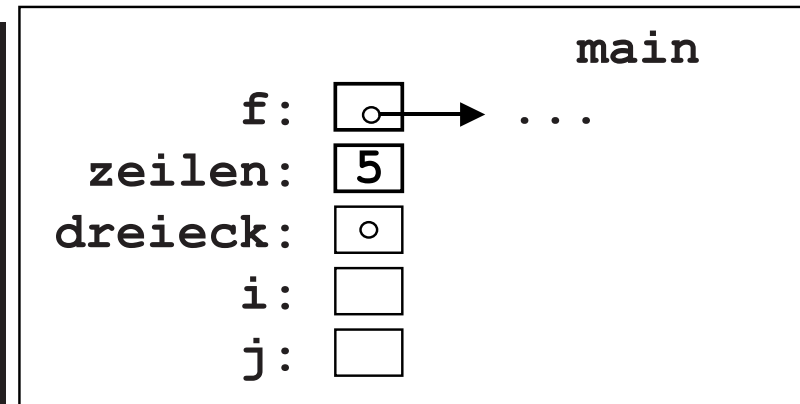
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

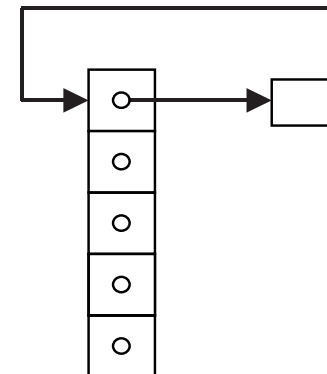
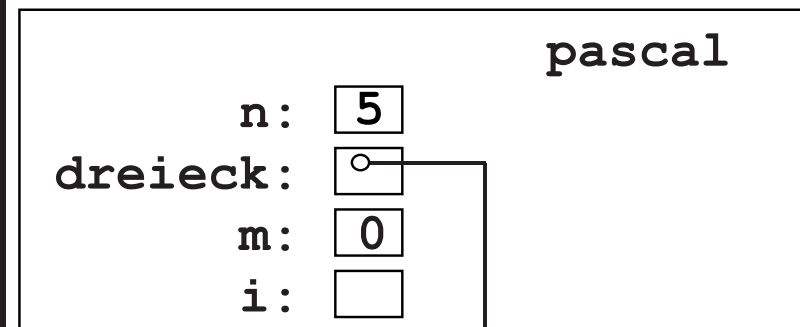
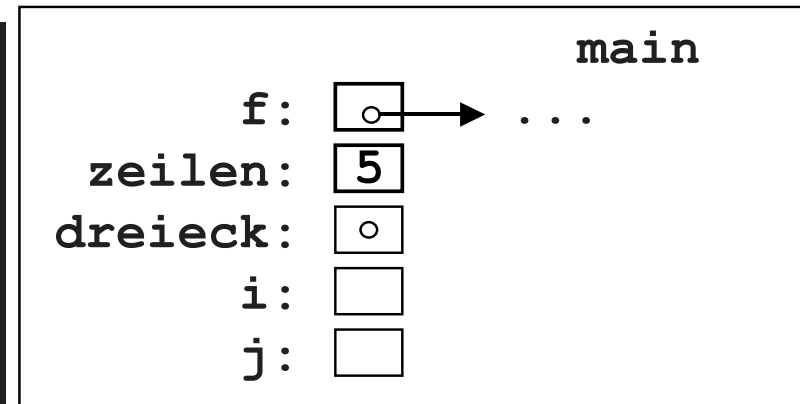
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

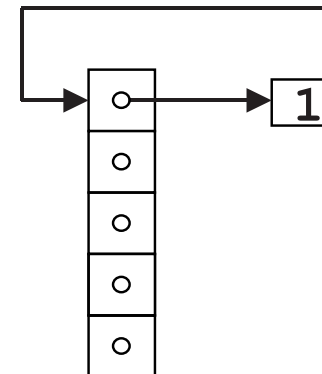
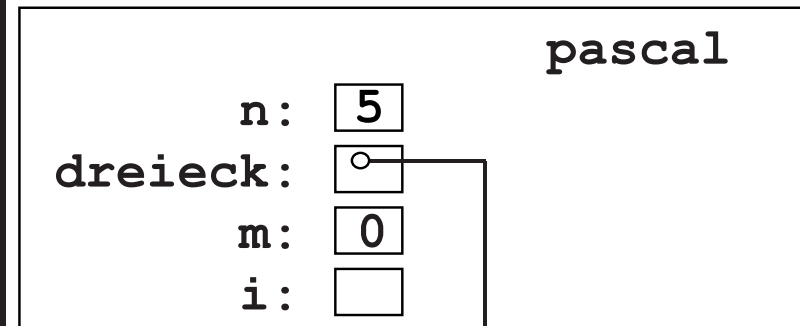
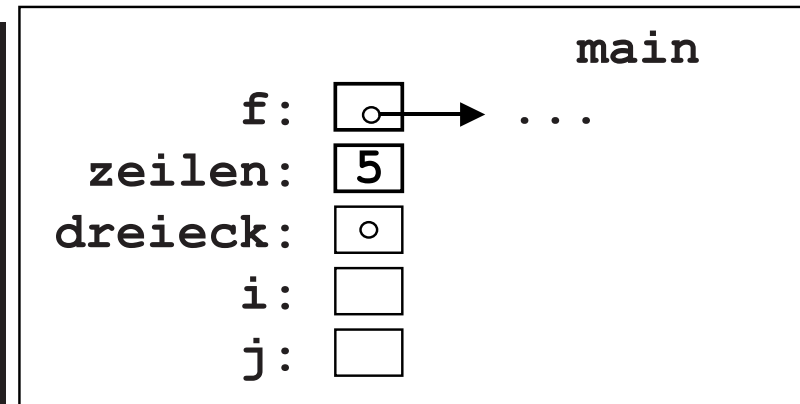
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

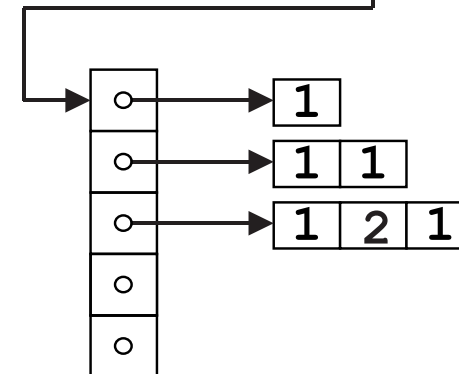
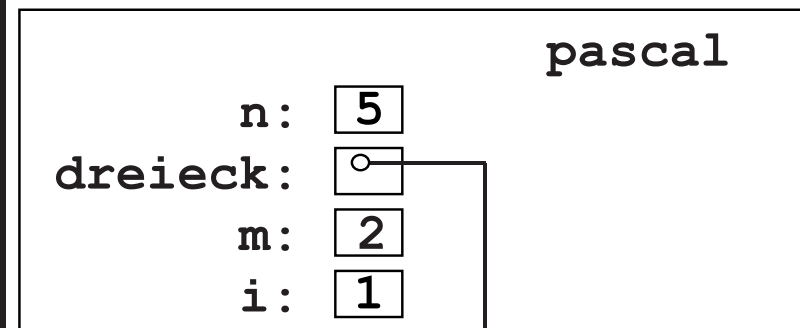
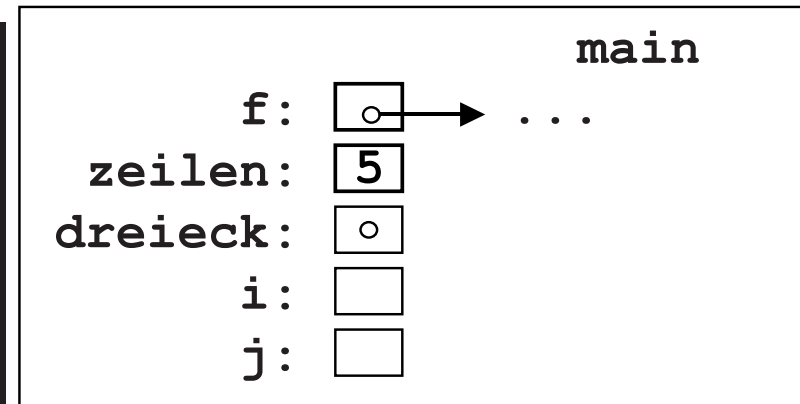
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



# Ablauf des Programms

```
public static void main(...) {
    Funktionsdienst f =
        new Funktionsdienst();
    int zeilen = 5;
    int[][] dreieck = f.pascal(zeilen);
    for (int i=0; i<zeilen; i++)
        for (int j=0; j<i+1; j++)
            System.out.println(dreieck[i][j]);
}
```

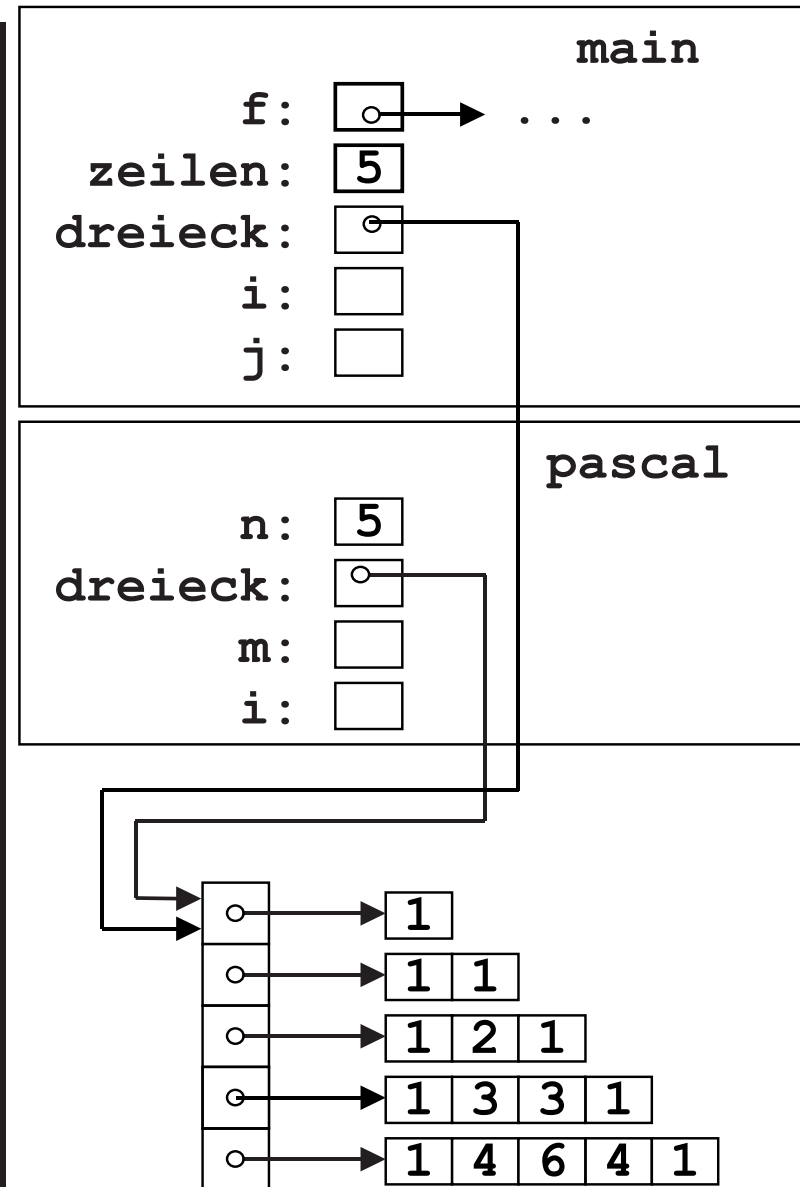
```
public int[][] pascal(int n) {
    int[][] dreieck = new int[n][];
    for (int m=0; m<n; m++) {
        dreieck[m] = new int[m+1];
        dreieck[m][0] = 1;
        dreieck[m][m] = 1;
        for (int i=1; i<m; i++)
            dreieck[m][i] = dreieck[m-1][i-1]
                + dreieck[m-1][i];
    }
    return dreieck;
}
```



# Ablauf des Programms

```
public static void main(...) {
    Funktionsdienst f =
        new Funktionsdienst();
    int zeilen = 5;
    int[][] dreieck = f.pascal(zeilen);
    for (int i=0; i<zeilen; i++)
        for (int j=0; j<i+1; j++)
            System.out.println(dreieck[i][j]);
}
```

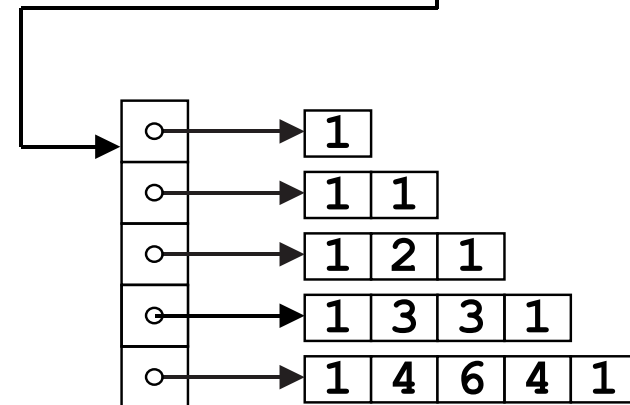
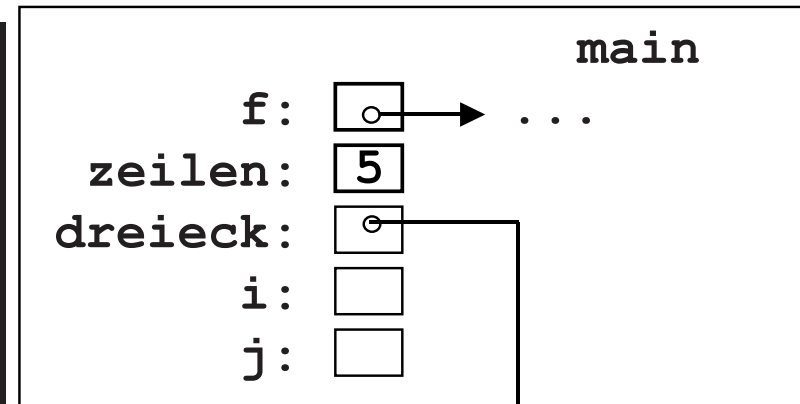
```
public int[][] pascal(int n) {
    int[][] dreieck = new int[n][];
    for (int m=0; m<n; m++) {
        dreieck[m] = new int[m+1];
        dreieck[m][0] = 1;
        dreieck[m][m] = 1;
        for (int i=1; i<m; i++)
            dreieck[m][i] = dreieck[m-1][i-1]
                + dreieck[m-1][i];
    }
    return dreieck;
}
```



# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

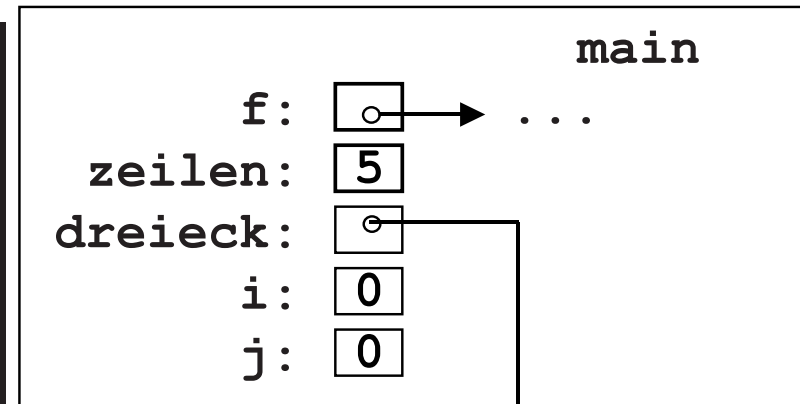
```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



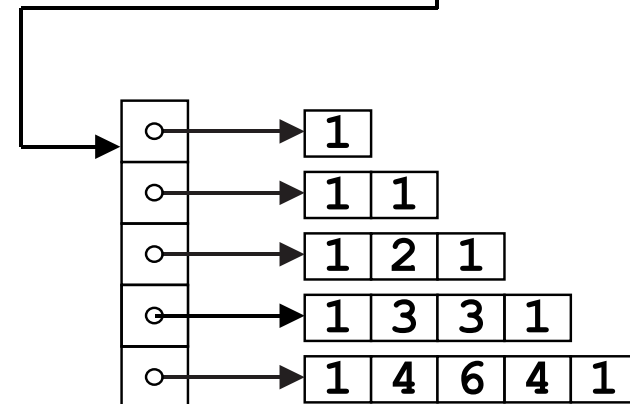
# Ablauf des Programms

```
public static void main(...) {  
    Funktionsdienst f =  
        new Funktionsdienst();  
    int zeilen = 5;  
    int[][] dreieck = f.pascal(zeilen);  
    for (int i=0; i<zeilen; i++)  
        for (int j=0; j<i+1; j++)  
            System.out.println(dreieck[i][j]);  
}
```

```
public int[][] pascal(int n) {  
    int[][] dreieck = new int[n][];  
    for (int m=0; m<n; m++) {  
        dreieck[m] = new int[m+1];  
        dreieck[m][0] = 1;  
        dreieck[m][m] = 1;  
        for (int i=1; i<m; i++)  
            dreieck[m][i] = dreieck[m-1][i-1]  
                + dreieck[m-1][i];  
    }  
    return dreieck;  
}
```



und so weiter  
und so weiter ...



## *Aufgabe 40: Die Klasse Initialisierer*

```
class Initialisierer {  
  
    Initialisierer() {}  
  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
        return;  
    }  
}
```

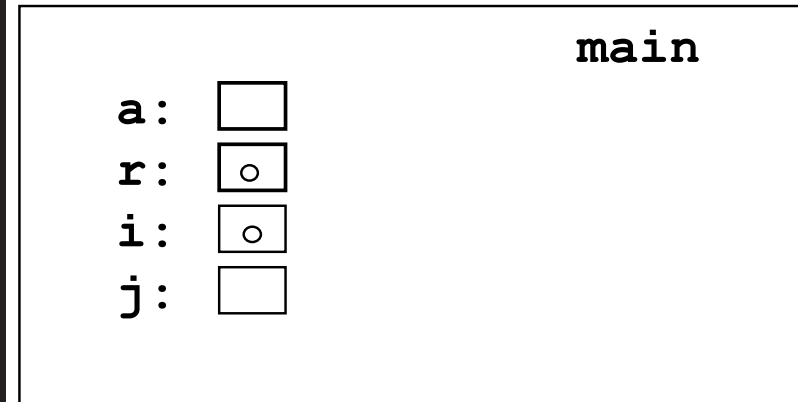
## *Aufgabe 40: Die Klasse Aufgabe40*

```
class Aufgabe40 {  
  
    public static void main(String[] args) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i = new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

# Das Programm aus Aufgabe 40

```
class Aufgabe40 {  
    public static void main(...) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i =  
            new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

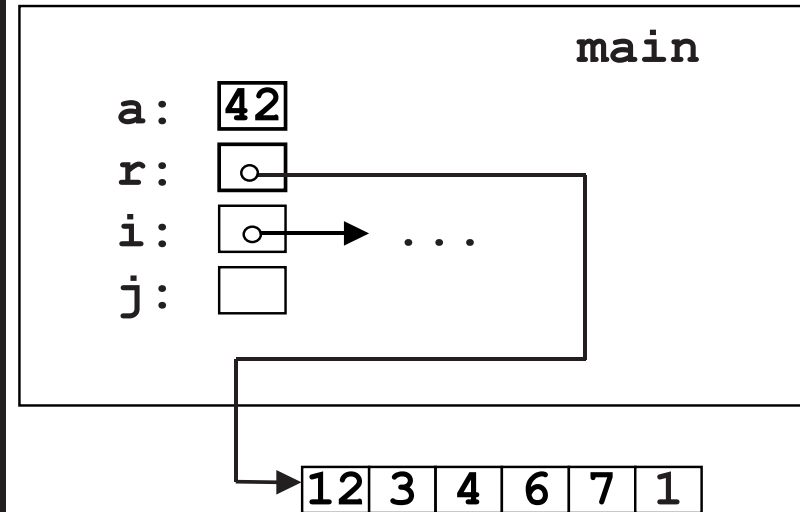
```
class Initialisierer {  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
        return;  
    }  
}
```



# Das Programm aus Aufgabe 40

```
class Aufgabe40 {  
    public static void main(...) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i =  
            new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

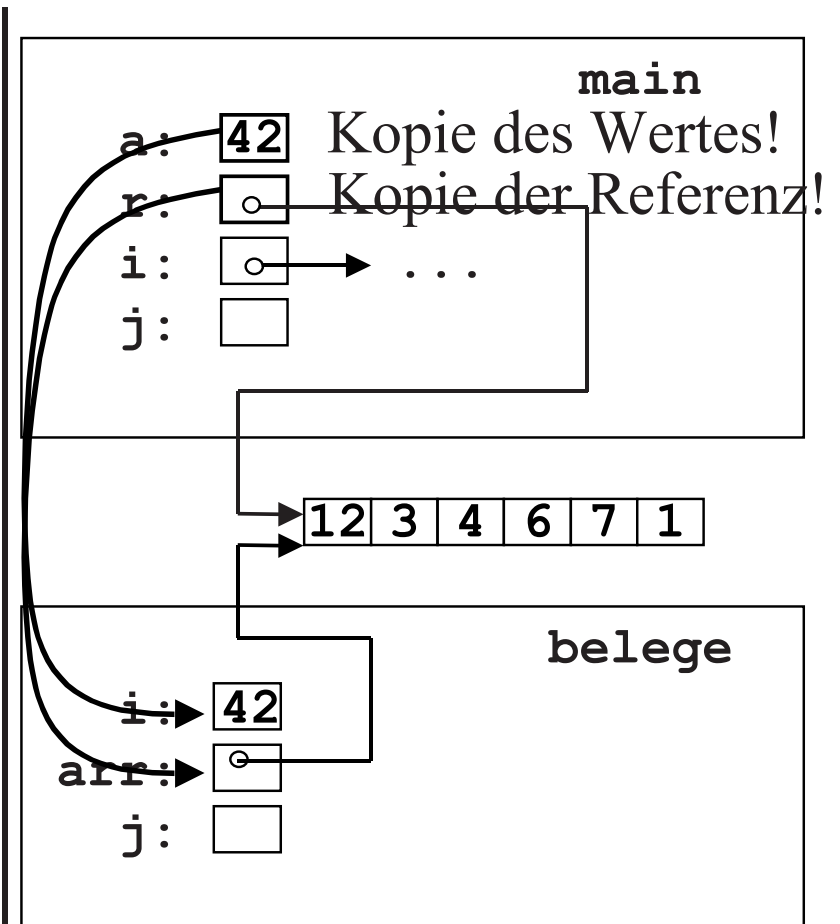
```
class Initialisierer {  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
        return;  
    }  
}
```



# Das Programm aus Aufgabe 40

```
class Aufgabe40 {  
    public static void main(...) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i =  
            new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

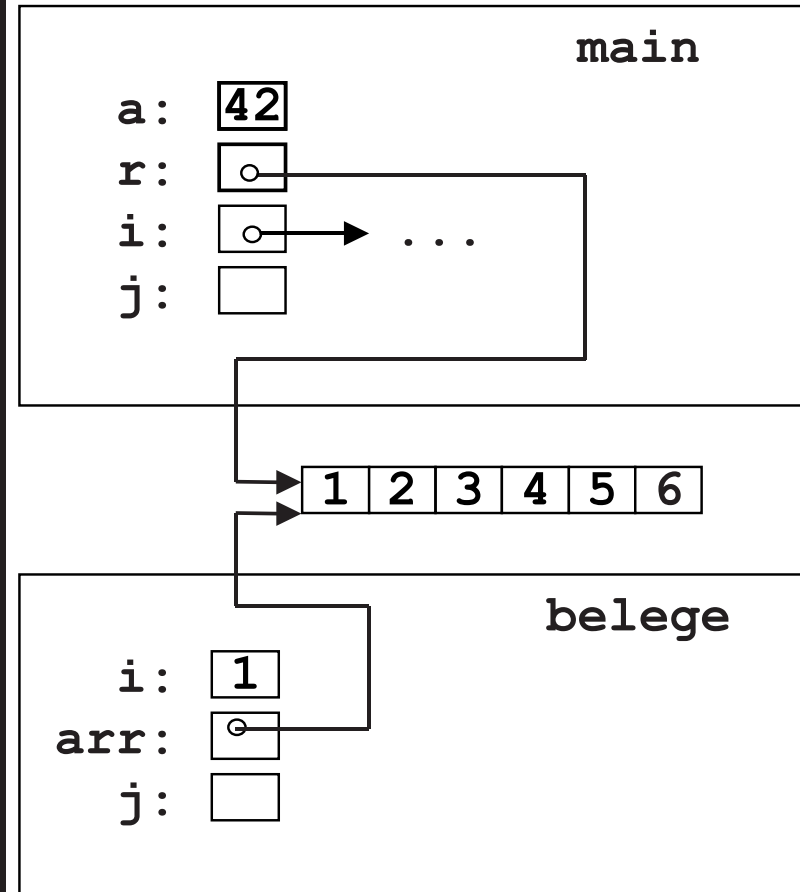
```
class Initialisierer {  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
        return;  
    }  
}
```



# Das Programm aus Aufgabe 40

```
class Aufgabe40 {  
    public static void main(...) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i =  
            new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

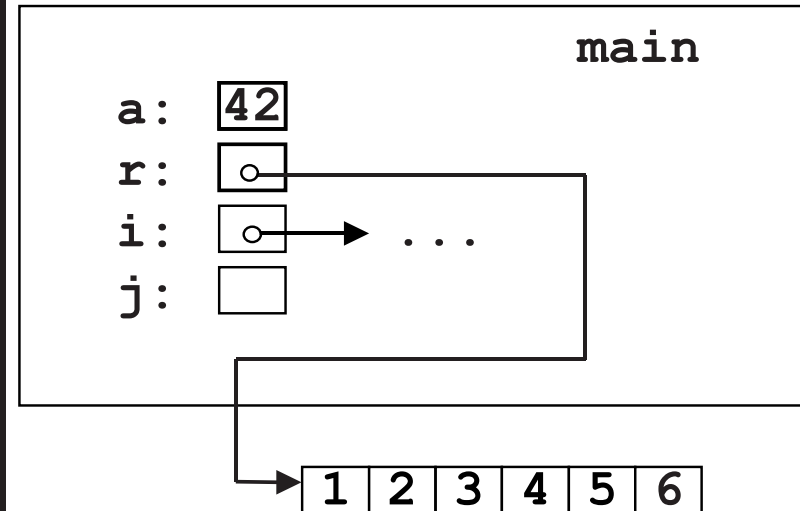
```
class Initialisierer {  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
    }  
}
```



# Das Programm aus Aufgabe 40

```
class Aufgabe40 {  
    public static void main(...) {  
        int a = 42;  
        int[] r = {12, 3, 4, 6, 7, 1};  
        Initialisierer i =  
            new Initialisierer();  
        i.belege(a, r);  
        System.out.println(a);  
        for (int j=0; j<r.length; j++)  
            System.out.println(r[j]);  
    }  
}
```

```
class Initialisierer {  
    void belege(int i, int[] arr) {  
        i = 1;  
        for (int j=0; j<arr.length; j++) {  
            arr[j] = j+1;  
        }  
        return;  
    }  
}
```



und so weiter  
und so weiter ...

**Vielen Dank für die  
Aufmerksamkeit!**