

Code & Material

3. Stunde, 31. Okt. 2012

Georg Trogemann

symbolisches vs. gegenständlichem Rechnen

Tonbörse mit 48 Kügelchen, Alter ca. 3.500 Jahre, Fund aus dem Jahre 1928/29,
(in: Georges Ifrah, *Universalgeschichte der Zahlen*)

Gegenstände, Hammel und Ziegen betreffend

- 21 Mutterschafe
- 6 weibliche Lämmer
- 8 erwachsene Hammel
- 4 männliche Lämmer
- 6 Mutterziegen
- 1 Bock
- (2) Jungziegen

*Abb. 3: Eiförmige Tonbörse (46 mm × 62 mm × 50 mm), entdeckt in den Ruinen des Palastes von Nuzi (mesopotamische Stadt; ca. 15. Jh. v. Chr.).
(Harvard Semitic Museum, Cambridge. Katalognummer SMN 1854)*



- Computertechnik ist ebenfalls gegenständliches Rechnen. An der Oberfläche wird symbolisches Rechnen simuliert. Doch die beiden Darstellungen sind miteinander verbunden. Wir haben es mit zwei Maschinen zu tun, der symbolischen und der physikalischen.

Geschichtliche Entwicklung: Mathematik

Der Methodenstreit

- Holzschnitt von Gregor Reisch aus dem Jahre 1504
- Die Göttin Arithmetica überwacht einen Wettbewerb zwischen dem Philosophen Boethius, der zu dieser Zeit irrtümlich für die Erfindung des Ziffernrechnens stand, und Pythagoras, der – ebenfalls irrtümlich – als Erfinder des Abacus-Rechnens galt. Der zufriedene Boethius ist bereits fertig, während Pythagoras noch missmutig rechnet. Arithmetica ist offensichtlich der neuen Technik zugeneigt.



Datentypen

- Integer

deklarieren: `int i; // ganzzahliger Wert zwischen
// zwischen -32.768 und 32.767 (2 Byte)`

benutzen: `i = 12;`

- Float

deklarieren: `float f; // (4 Byte)`

benutzen: `f = 1.532;`

- Boolean

deklarieren: `boolean digValue;`

benutzen: `digValue = true; // Kann nur true oder false sein.`

Typenumwandlung (cast)

```
int x;  
int y;  
float z;
```

```
x = 1;  
y = x / 2;          // y now contains 0, ints can't hold  
    fractions  
z = (float)x / 2.0; // z now contains .5 (you have  
    // to use 2.0, not 2)
```

If/else-Abfrage

```
// Es wird der Wert in den Klammern ausgewertet.  
// Wenn der Wert true ist, werden der nächste Block  
// (die Befehle zwischen den geschweiften  
// Klammern) ausgeführt, ansonsten (d.h. wenn er  
// false ist) nicht.
```

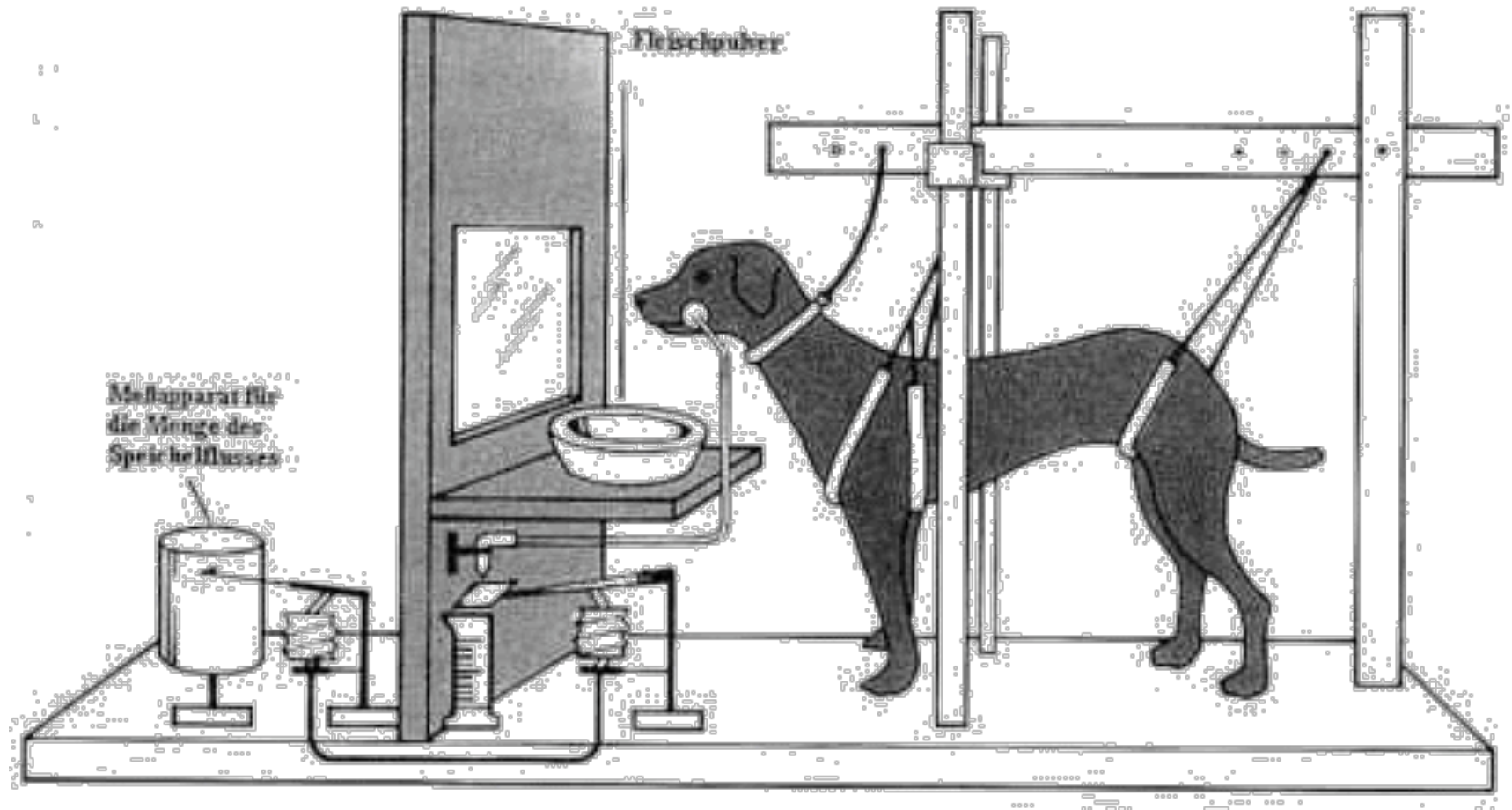
```
if (this question is true) {  
    // do Thing A  
} else if (this question is true) {  
    // do Thing B  
} else {  
    // do Thing C  
}
```

Vergleichsoperatoren

Folgende Terme sind möglich sowie beliebige Kombinationen, die mit den Booleschen Operatoren `&&` (and), `||` (or) und `!` (not) gebildet werden können.

- `x == y` (x is equal to y)
- `x != y` (x is not equal to y)
- `x < y` (x is less than y)
- `x > y` (x is greater than y)
- `x <= y` (x is less than or equal to y)
- `x >= y` (x is greater than or equal to y)

Technische Reiz-Reaktions-Systeme



Pawlows Hund