

Processing.org workshops

Workshop 3

Open Space Aarhus

Bryggervej 30, 8240 Århus N

30. november 2011



Dagens program

- Introduktion
- Resume af sidste gang
- Kode
 - Løkker
 - Arrays
 - Funktioner
- Workshop
 - Mange bolde på en gang!
 - Fyrværkeri (fontæne)
- *Afslutning*

Hvad har de flittige lavet

- `http://www.openprocessing.org/classrooms/?classroomID=1075`

Slides og processing filer

<http://poodle/processing>

Slides kan sikkert bruges til at kigge i eller kopiere fra.

- Variable: `float boldX = 200;`
- Operationer: `boldX = boldX + deltaX;`
- Løkker:
 - `while (betingelse) { ... }`
 - `for (start; betingelse; opdatering;) { ... }`
- Forgreninger:
 - `if (x > width) { dx = -dx; }`
 - `if (betingelse) { ... }`
 - `if (betingelse) { ... } else { ... }`
- Simpel fysik.
 - `acceleration = summen af kræfter delt med partiklens masse`
 - `ny hastighed = gammel hastighed + acceleration`
 - `ny position = gammel position + hastighed`
- Spørgsmål?

Løkker

Lav en løkke som tegner 10 bolde ved siden af hinanden. Brug x som tæller

med while-løkken

```
int x = 0;
while (x < 10) {
  ellipse(20 + x * 40, 200, 30, 30);
  x += 1;
}
```

med for-løkken

```
for (int x = 0; x < 10; x += 1) {
  ellipse(20 + x * 40, 200, 30, 30);
}
```


Arrays

Et array er en opslagstabel. Man kan lave et array med et fast antal pladser. Derefter kan man skrive og læse værdier på de enkelte pladser i arrayet.

- deklaration: `float[] boldX = new float[10];`
- tildeling: `boldX[5] = 100;`
- læsning: `ellipse(boldX[5], boldY[5], 30, 30);`

Skal vi have en lille øvelse med arrays?

Arrays til mange bolde

- `float[] boldX = new float[10];`
- `float[] boldY = new float[10];`
- `float[] deltaX = new float[10];`
- `float[] deltaY = new float[10];`

Arrays og Løkker

Arrays og løkker er som skabt til hinanden. Vi kan bruge en for-løkke til at løbe igennem kuglerne og opdatere deres positioner.

```
int ANTAL = 10;

float[] kugleX = new float[ANTAL];
float[] kugleY = new float[ANTAL];
float[] deltaX = new float[ANTAL];
float[] deltaY = new float[ANTAL];

for (int i = 0; i < ANTAL; i++) {
    kugleX[i] += deltaX[i];
    kugleY[i] += deltaY[i];
}
```

Kanoneksemplet udvidet til at skyde med flere kanonkugler samtidigt.

<http://www.openprocessing.org/visuals/?visualID=47076>

Funktioner

Små genbrugelige stumper kode. Også nyttig til at gøre koden mere overskuelig.

Du har allerede brugt en masse funktioner fra processing. Du har også skrevet dine egne fx. `setup()` og `draw()`. Nu vil vi lave vore helt egne.

En funktion:

- kan tage input via parametre
- kan returnere et output
- kan have sideeffekter (f.eks. tegne på skærmen)

Funktioner

```
returtype navn(parametre) {  
  // implementation  
}
```

Eksempler

```
float doubleUp(float value) {  
  return value * 2;  
}
```

```
void draw() {  
}
```

```
float sumOf(float a, float b) {  
  return a + b;  
}
```

Tak for i dag

- Hvad syntes *du* om i dag?
- Næste gang: ?
- T^3 i må meget gerne hjælpe med at rydde lokalet.

Klasseværelset

www.openprocessing.org/classrooms/?classroomID=1075