

# Programmeringsopgave PacMan

IT Camp for piger 2023

<b>Opsætning af Greenfoot</b>	<b>2</b>
Windows	2
Mac	2
Linux	7
Importer Pacman startkode	7
Eksempel på ændring i koden / Intro til greenfoot	8
<b>Opgaver</b>	<b>12</b>
Ekstra opgaver - vælg en af gangen	12
<b>Hints til brug af Pacman og Greenfoot API'en</b>	<b>13</b>



# Opsætning af Greenfoot

- Greenfoot er det program hvormed den kode som I får givet skal åbnes. Derfor skal I også lige starte ud med at downloade det :-)






## Windows

Hvis du kører Windows kan Greenfoot anskaffes på følgende måde:

1. Gå til <https://www.greenfoot.org/download>
2. Klik "Standalone", for at downloade standalone windows udgaven

Version: 3.6.1 (fixes several notable bugs, including a lock-up), 6 Dec 2019

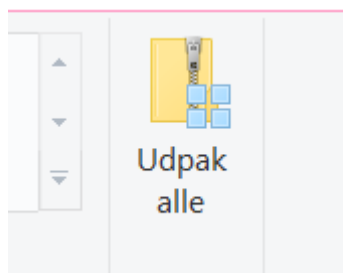
Installation Instructions can be found [here](#) and frequently asked questions are answered [here](#).

Windows	Mac OS X	Ubuntu	Pure Java	Standalone
7, 8 and 10.	10.10 or later.	For 64-bit x86 CPUs. Supports Debian.	For any OS. Requires JDK.	For Windows, runs from a USB stick. Includes JDK.
				

There is also a Debian/Ubuntu package for other CPUs (e.g. Raspberry Pi) [available](#).

**Note:** Greenfoot 3.6.0 onwards on Intel/AMD processors requires a 64-bit CPU and 64-bit operating system. For 32-bit support, use Greenfoot 3.5.4 from our [old versions page](#).

3. Når download er færdigt, udpak hele greenfoot standalone zip filen til et passende sted.



4. Åben den udpakkede folder, og kørs greenfoot.exe
5. Greenfoot kan nu bruges

Note: Greenfoot kan også installeres gennem Windows installeren, hvis man ønsker at bruge det i længere tid.

## Mac






Hvis du kører MacOS kan Greenfoot anskaffes på følgende måde:

1. Gå til <https://www.greenfoot.org/download>

2. Klik "MAC OS X", for at downloade mac udgaven

Version: 3.6.1 (fixes several notable bugs, including a lock-up), 6 Dec 2019

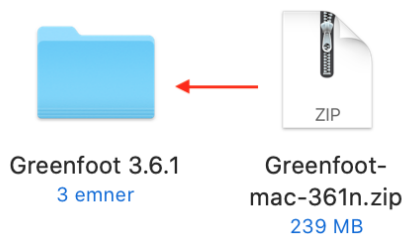
Installation Instructions can be found [here](#) and frequently asked questions are answered [here](#).

Windows	Mac OS X	Ubuntu	Pure Java	Standalone
7, 8 and 10.	10.10 or later.	For 64-bit x86 CPUs. Supports Debian.	For any OS. Requires JDK.	For Windows, runs from a USB stick. Includes JDK.
				

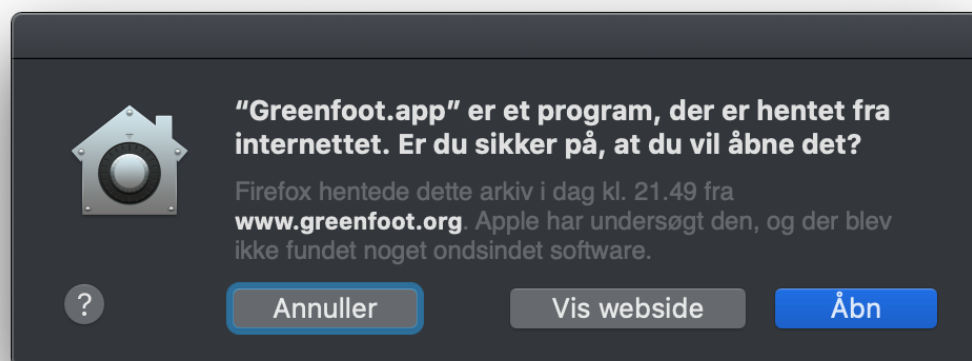
There is also a Debian/Ubuntu package for other CPUs (e.g. Raspberry Pi) [available](#).

**Note:** Greenfoot 3.6.0 onwards on Intel/AMD processors requires a 64-bit CPU and 64-bit operating system. For 32-bit support, use Greenfoot 3.5.4 from our [old versions page](#).

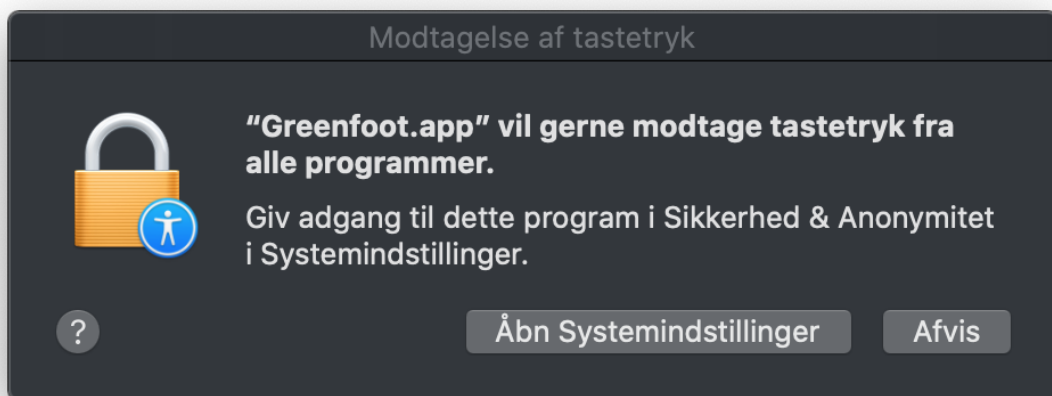
3. Når download er færdig dobbelt-klikker du på den downloadede .zip fil for at udpakke den.



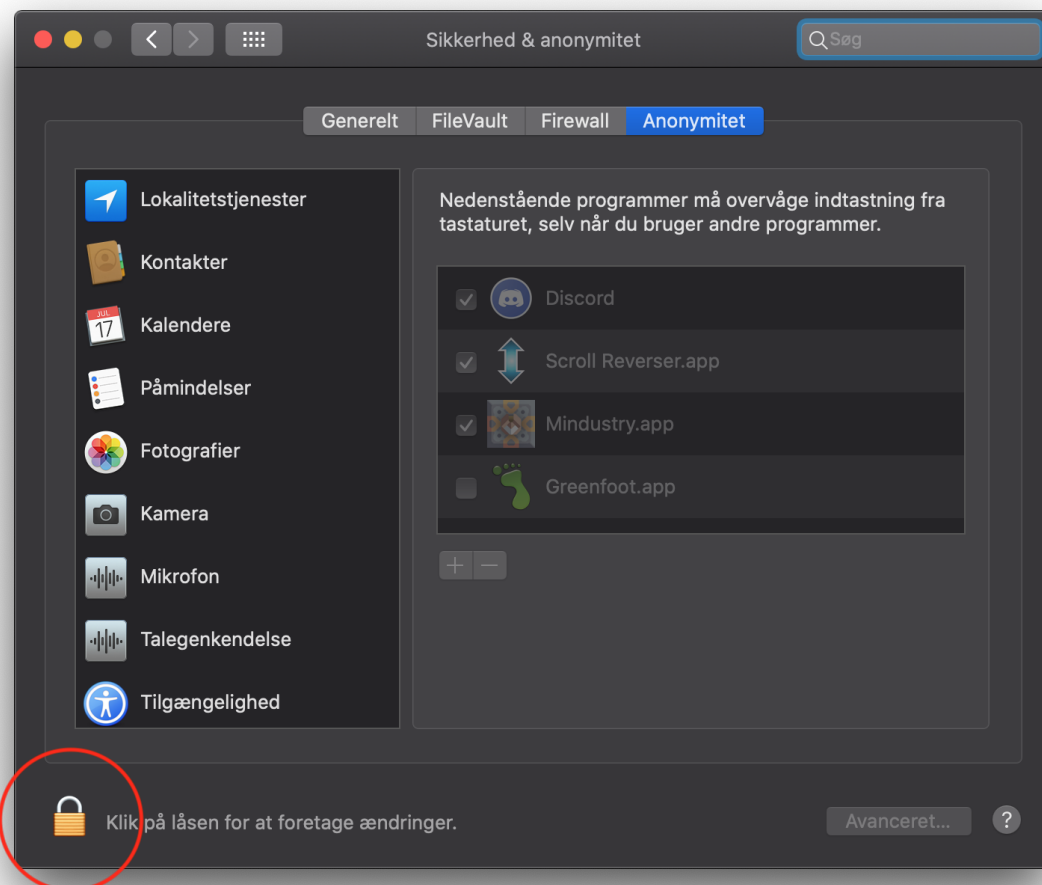
4. Åben den udpakkede mappe og kør Greenfoot.app
5. Hvis du kører en af de nyeste versioner af macOS, vil din computer klage over at Greenfoot er hentet fra internettet. Tryk "Åbn" for at åbne programmet:



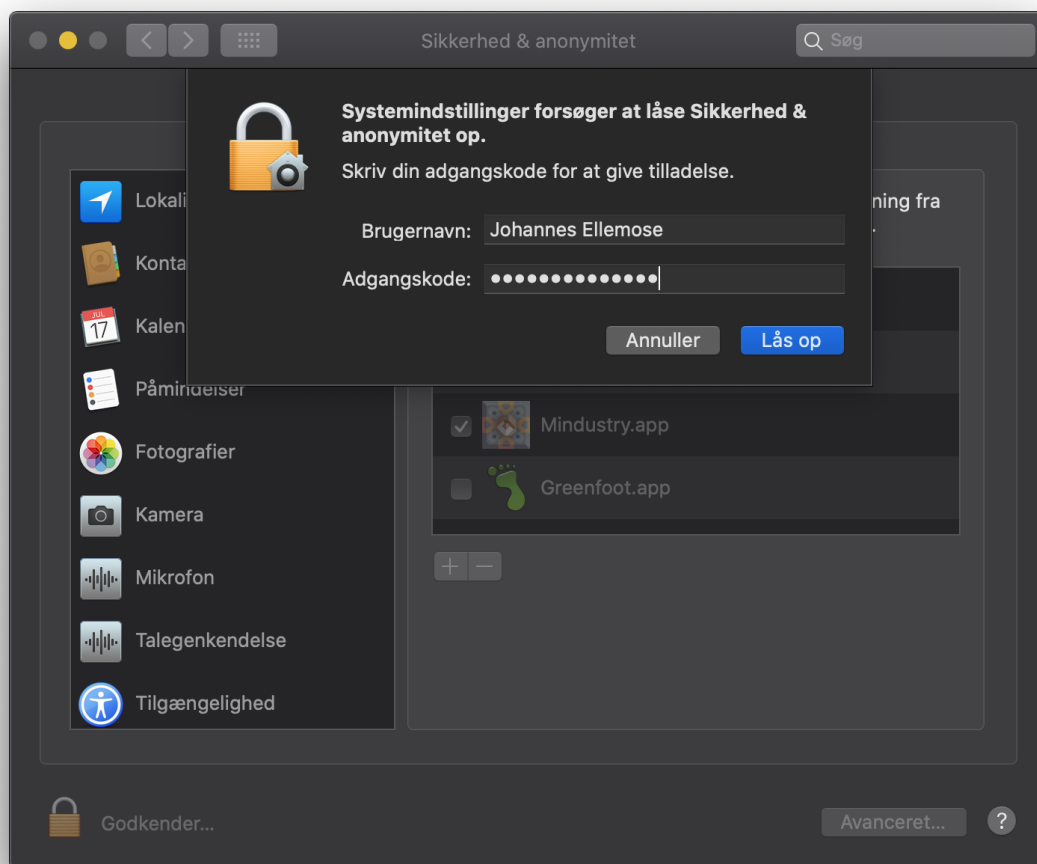
6. Ligeledes spørger macOS muligvis om du vil tillade Greenfoot at modtage tastetryk. Dette kan være nødvendigt for at programmet fungerer korrekt. Tryk "Åbn Systemindstillinger" for at tillade dette:



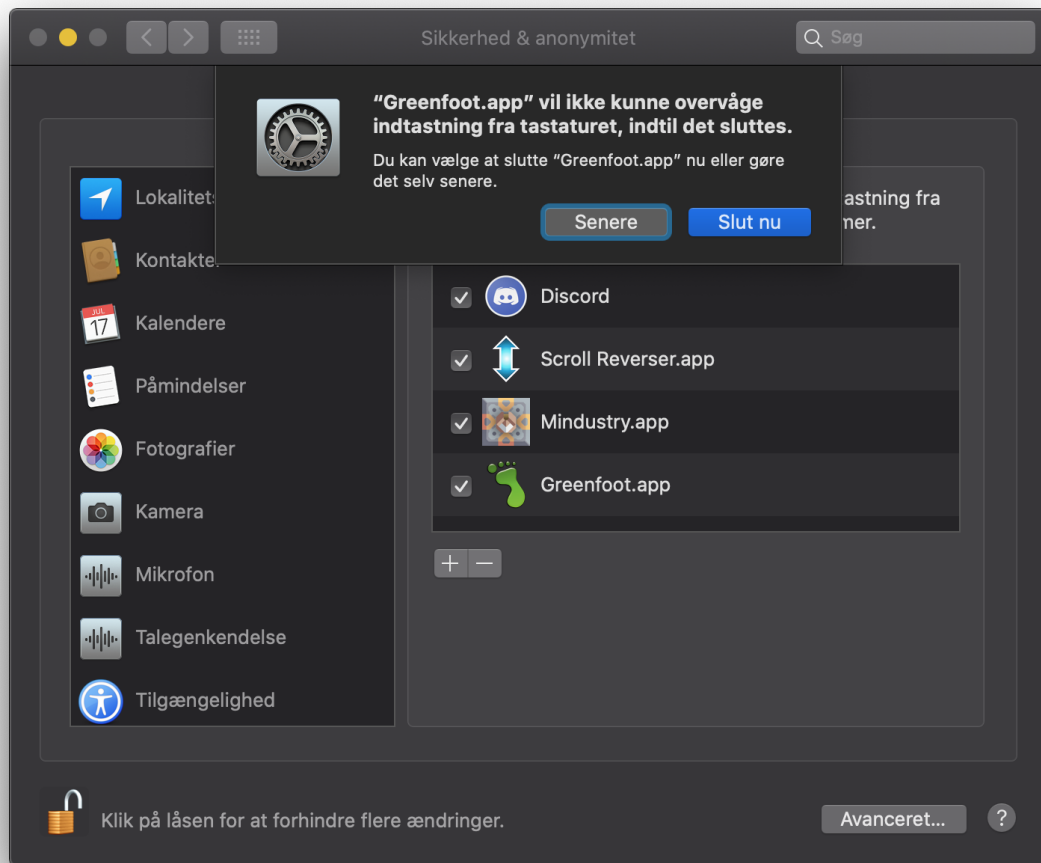
7. Tryk på låsen nederst til venstre for at låse op for ændringer:



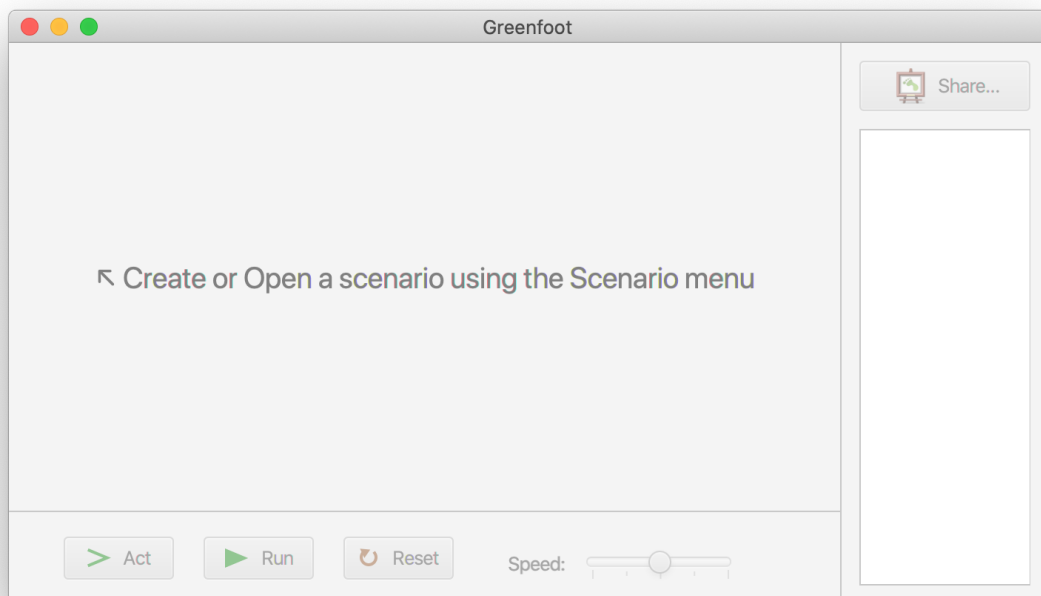
8. Skriv din computers adgangskode og tryk på "Lås op":



9. Nu kan du tjekke boksen af ud for Greenfoot. En besked popper op og fortæller at Greenfoot skal slutes for at ændringerne kan træde i kraft. Tryk på "Slut nu" for at afslutte Greenfoot. Tryk evt. på låsen for at låse ændringer:



10. Kør Greenfoot.app igen. Denne gang skulle der ikke være nogle pop-up beskeder:



## Linux

Hvis du kører en Linux distribution er der god chance for at du har styr på hvordan man anskaffer greenfoot.

For Ubuntu, Debian og lignende har Greenfoot selv en installationsguide på

<https://www.greenfoot.org/download/installation>

For Archlinux findes AUR pakken [greenfoot](#)

For andre distributioner stoler vi på du ved hvad du laver.

# Importer Pacman startkode

1. Download startkoden fra:  
[https://drive.google.com/file/d/182Dftb5\\_ZQpR-Rrbu7j8X8Yb2L81HtUE/view](https://drive.google.com/file/d/182Dftb5_ZQpR-Rrbu7j8X8Yb2L81HtUE/view)
2. Udpak zip filen til et passende sted
3. I Greenfoot fra menubaren, vælg "Scenario" → "Open" → Naviger til folderen du pakkede zip filen ud i, og vælg "Open"
4. Greenfoot skulle nu gerne have åbnet Pacman projektet.



## Eksempel på ændring i koden / Intro til greenfoot

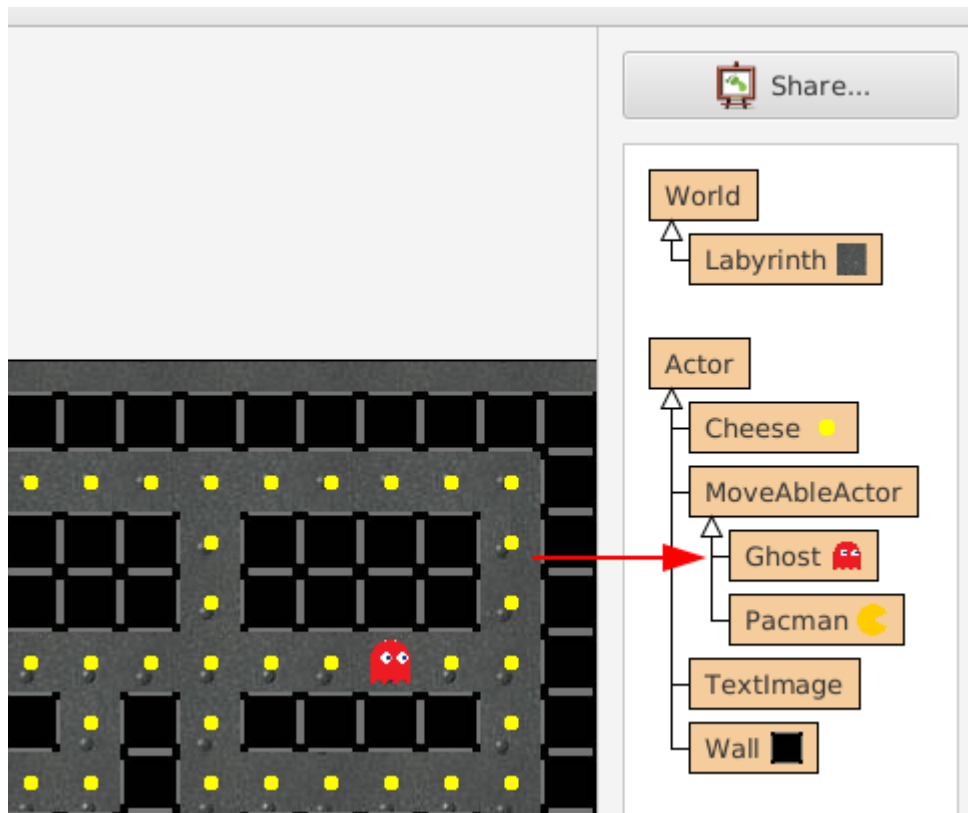
For at demonstrere workflowet i Greenfoot laver vi her en ændring i koden. Denne ændring er inkluderet i det downloadede projekt.

Lige nu står spøgelserne helt stille når spillet kører. Vi vil gerne give dem en simpel opførsel hvor

1. Spøgelset bevæger sig lige ud indtil det rammer en væg
2. Når det rammer en væg skifter den til en tilfældig ny retning.

Lad os lave den ændring.

Vi åbner Ghost klassen ved at dobbeltklikke på dens ikon i klassesdiagrammet



Med dette åbner editoren med Ghost klassen

```
Class Edit Tools Options
Ghost X
Compile Undo Cut Copy Paste Find... Close Source Code
import greenfoot.*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/**
 * Ghosts sole purpose in life is to kill Pacman and thereby protect their precious cheese.
 * They move randomly around the labyrinth, while looking for a chance to kill Pacman.
 *
 * @author Camilla Marie Vinther Frederiksen
 * @version 2, 01-10-2016
 */
public class Ghost extends MoveAbleActor
{
    /**
     * Act - move randomly around the world without colliding with the walls.
     * If touching Pacman, then kill him.
     */
    public void act() {
        killPacman();
    }

    /**
     * If touching Pacman, kill him and end the game!
     */
    private void killPacman() {
        //Write code to kill Pacman and end game
    }
}
```

For at tilføje den ønskede opførsel tilføjer vi et nyt metodekald i `act()`. `act()` bliver kaldt hver gang "Act" knappen i Greenfoot vinduet bliver klikket, eller kontinuerligt hvis "Run" knappen er blevet klikket.

Vi tilføjer et kald til `act()`, kaldet `moveRandom()`

```
public void act() {
    moveRandom();
    killPacman();
}
```

Greenfoot beklager sig fordi metoden ikke findes, så vi opretter den nedenunder funktionen `killPacman()`

```
private void moveRandom() {
}

}
```

Vi kan nu begynde at implementere funktionaliteten.

1. Spøgelset bevæger sig lige ud indtil det rammer en væg

Vi laver i `moveRandom()` et kald til metoden `moveInDirectionIfPossible()`, som er defineret i `MoveableActor`. `moveInDirectionIfPossible()` tager en parameter som er retningen actoren skal forsøge at bevæge sig i. Siden vi gerne vil bevæge os i den samme retning husker vi dette i en feltvariabel som vi kalder `currentDirection`

```
public class Ghost extends MoveableActor
{
    private int currentDirection = RIGHT;

    private void moveRandom() {
        moveInDirectionIfPossible(currentDirection);
    }
}
```

Hvis vi klikker "Run" nu vil alle spøgelserne begynde at bevæge sig mod højre indtil de rammer en mur.

Vi kan nu tilføje den anden del af opførelsen

2. Når det rammer en væg skifter den til en tilfældig ny retning.

Hvis spøgelset rammer en mur kan det ikke længere bevæge sig i den retning.

Metoden `moveInDirectionIfPossible()` returnerer en boolean (true/false) om det lykkedes at bevæge sig. Den kan vi bruge til at vurdere om en ny retning er nødvendig.

Vi opbevarer om bevægelse lykkedes i en lokal boolean variabel

```
private void moveRandom() {
    boolean hasMoved = moveInDirectionIfPossible(currentDirection);
}
```

Og vi kan nu tilføje en if sætning

```
private void moveRandom() {
    boolean hasMoved = moveInDirectionIfPossible(currentDirection);
    if (!hasMoved) {
        // Turn in a new direction
    }
}
```

If sætninger fungerer sådan, at hvis indholdet i parenteserne evalueres til sandt, så bliver koden i tuborg-parenthesen kørt. "!" er et udtryk som betyder "ikke". Altså, hvis det **ikke** lykkedes at bevæge sig, skal vi køre det indre kode. Hvis dette sker skal vi sætte en ny tilfældig retning.

Til at vælge en tilfældig retning kan vi bruge [Greenfoot.getRandomNumber\(int limit\)](#), som returnerer et tilfældigt tal fra 0 (inkludativ), op til, men ikke inkluderende *limit*.

Vi vil gerne have en tilfældig retning ud af fire, alle i en 90 graders vinkel. Derfor kan vi skrive:

```
private void moveRandom() {  
    boolean hasMoved = moveInDirectionIfPossible(currentDirection);  
    if (!hasMoved) {  
        currentDirection = (Greenfoot.getRandomNumber(4) * 90);  
    }  
}
```

Hvis vi nu klikker "Run" opfører spøgelseerne som forventet. De bevæger sig til de rammer en væg, skifter retning, og bevæger sig så igen.

# Opgaver

1. Styr Pacman med piletaster
2. Få Pacman til at spise ost
3. Få spøgelser til at dræbe Pacman

## Ekstra opgaver - vælg en af gangen

Lav to-player spil

1. Styr Pacman med andre taster
2. Fjern alle spøgelser - undtagen ét
3. Styr spøgelset med andre piletaster
4. Udskriv hvem der vinder

Lav labyrinten om

1. Byt rundt på væggene
2. Placer ostene et andet sted

Kom lyde på spillet

1. Lav din egen lyd fil eller hent en fra nettet  
HINT: Greenfoot har indbygget optager (*tools>Show Sound Recorder*)
2. Afspil en lyd - fx når Pacman spiser en ost  
HINT: Greenfoot har en lyd-klasse; *GreenfootSound*

Få Pacman til at bevæge sig hurtigere end Spøgelserne

1. Find den linje der faktisk bevæger Pacman (led i *MoveableActor*)
2. Find den linje der faktisk bevæger Spøgelserne (led i *MoveableActor*)
3. Find på en ændring, så Pacman går hurtigere end spøgelserne.

Lav noget du selv synes kunne være sjovt! Man kan nærmest lave ALT.

- fx lav om i billederne så Pacman får solbriller eller bliver til Ms. Pacman
-

# Hints til brug af Pacman og Greenfoot API'en

I spillet nedarver Pacman fra MoveableActor. Det betyder at Pacman klassen kan gøre brug af metoder fra MoveableActor. I denne klasse kunne følgende metoder være af interesse:

Metodenavn mm.	Forklaring
<code>endGameIfDone();</code>	Tjekker om alle oste er blevet spist eller Pacman ikke længere findes. I så fald viser en passende besked og stopper spillet.
<code>moveInDirectionIfPossible(UP);</code>	Tjekker om actoren (Pacman eller Ghost) kan bevæge sig i en bestemt retning, og bevæger sig så i den retning hvis det kan lade sig gøre. Returnerer om det lykkedes at bevæge sig. Tager en retning i grader som parameter. Her kan følgende konstanter bruges:
UP DOWN LEFT RIGHT	Konstanter som beskriver de fire mulige retninger en actor kan bevæge sig i spillet. Kan bruges som parameter, eks: <code>moveInDirectionIfPossible(DOWN);</code> Vinklen de holder er grader i urets retning, startende med pegende mod højre som 0 grader.

Følgende er nyttige metoder fra de forskellige klasser som Greenfoot tilgængeliggør:

Klassenavn.Metodenavn	Forklaring
<code><a href="#">Greenfoot.isKeyDown</a>("W");</code>	Tjekker om en bestemt tast på tastaturet er trykket ned. Almindelige bogstaver kan tjekkes med deres bogstav, eg. "S". Piletaster kan tjekkes med "UP", "LEFT" osv Eksempel: <pre>if (Greenfoot.isKeyDown("F")){     // Do thing }</pre>
<code><a href="#">removeTouching</a>(Cheese.class);</code>	Fjerner alle objekter af en given klasse, som det kaldende objekt berører. Eksempel: <pre>removeTouching(Wall.class); //Fjerner alle Wall objekter som den kaldende Actor berører</pre>
<code><a href="#">isTouching</a>(Cheese.class);</code>	Tjekker om den kaldende Actor berører et objekt af given klasse. Returnerer en boolean (true or

	false) Eksempel: <pre>if (isTouching(Wall.class)){     // Ran into a wall,     // play bonk noise. }</pre>
<a href="#">move(1);</a>	Bevæger en actor fremad en bestemt afstand Eksempel: <pre>move(2); // Move 2 units forward.</pre>
<a href="#">GreenfootSound</a>	En klasse som håndterer lyde, og afspilning af lyde. Eksempel: <pre>private GreenfootSound wallHitSound = new GreenfootSound("sounds/bonk.mp3");</pre>
<a href="#">GreenfootSound.play();</a>	Afspiller lyden som objektet holder. Eksempel: <pre>wallHitSound.play();</pre>

Hvis du ikke kan finde det du leder efter her, kan den fulde API findes på:

<https://www.greenfoot.org/files/javadoc/>

♥ IT Camp