

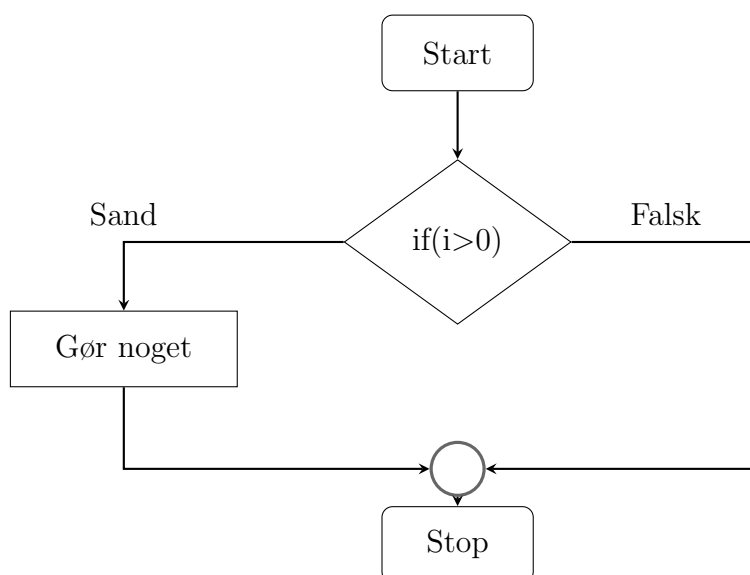
9 Betingelser/Forgrening

Læs om Betingelser/forgreninger i kapitel 3.2 i Systime bogen.
Se min video klik her: om betingelser

9.1 Betingelse if

```
if (betingelse , enten sand eller falsk) {  
    //kode  
}
```

Figur 15: if

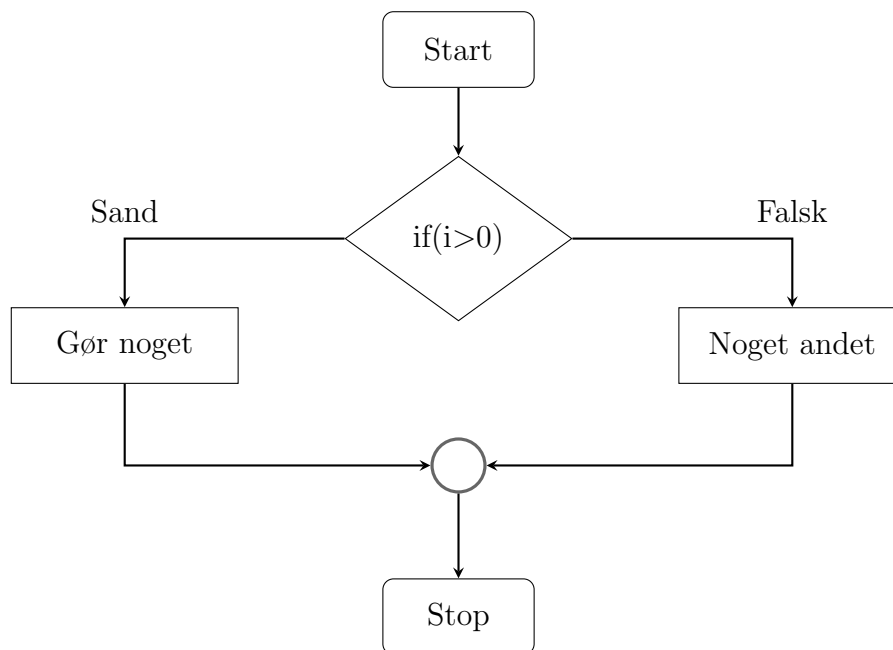


Figur 16: Rutediagram if

9.2 Forgrening if/else

```
if (betingelse , enten sand eller falsk) {  
    //kode  
}  
else{  
    //anden kode  
}
```

Figur 17: if/else

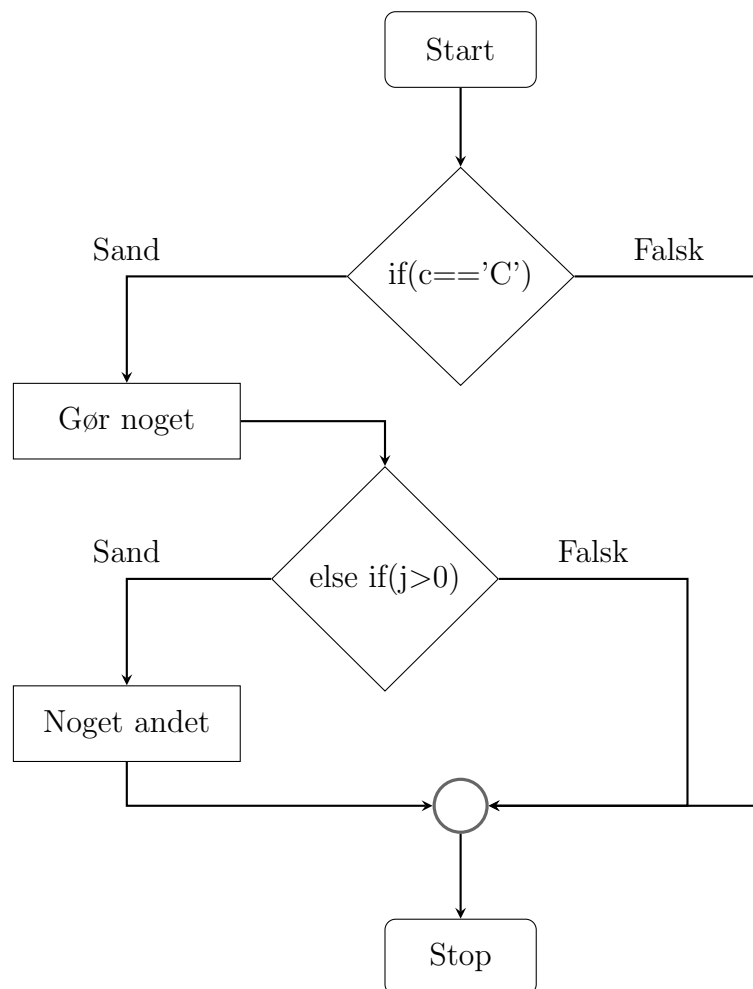


Figur 18: Rutediagram if/else

9.3 Forgrening if/else if

```
if (betingelse , enten sand eller falsk) {  
    // Din kode .  
}  
else if(ny betingelse , enten sand eller falsk) {  
    // Anden kode  
}
```

Figur 19: if/else if



Figur 20: Rutediagram if/else if

9.4 Opgave

Du skal lave et spil hvor du kan spille: Sten, papir, saks imod computeren. Sten vinder over saks, papir vinder over sten og saks vinder over sten. Der er i alt 9 forskellige kombinationer hvoraf tre er uafgjort.

Da spillet kræver et input, kan vi sige at '1'-tallet på tastaturet repræsenterer "sten", '2' = "papir", '3' = "saks". Brug funktionen `random(49, 52)`; til at generere computerens træk. Hvorfor 49,50,51? Jo for i processing er der en variable "key". Den variabel indeholder ASCII værdien for det sidste tastetryk.

```
'1'=49
```

```
'2'=50
```

```
'3'=51
```

Slå funktionen `random()` og variabelen `key` op i dokumentationen. Jeg vil foreslå at du benytter dig af `if/else if` som kan ses i figur: 19.

Start med det letteste tilfælde, når det er uafgjort: Hvis dit træk er det samme som computerens udskriv til skærmen "It's a tie"

```
// herefter arbejder vi os igennem resten ellers hvis computerens træk er  
sten
```

```
hvis du har saks
```

```
udskriv til skærmen "you lost!"
```

```
hvis du har papir
```

```
udskriv til skærmen "you win!"
```

Konstruer selv forgreningen hvis computerens træk er papir eller saks
Du kan finde hjælp i filen `helpMe.pde` på github.

Der er flere opgaver Codingbat. Logic 1