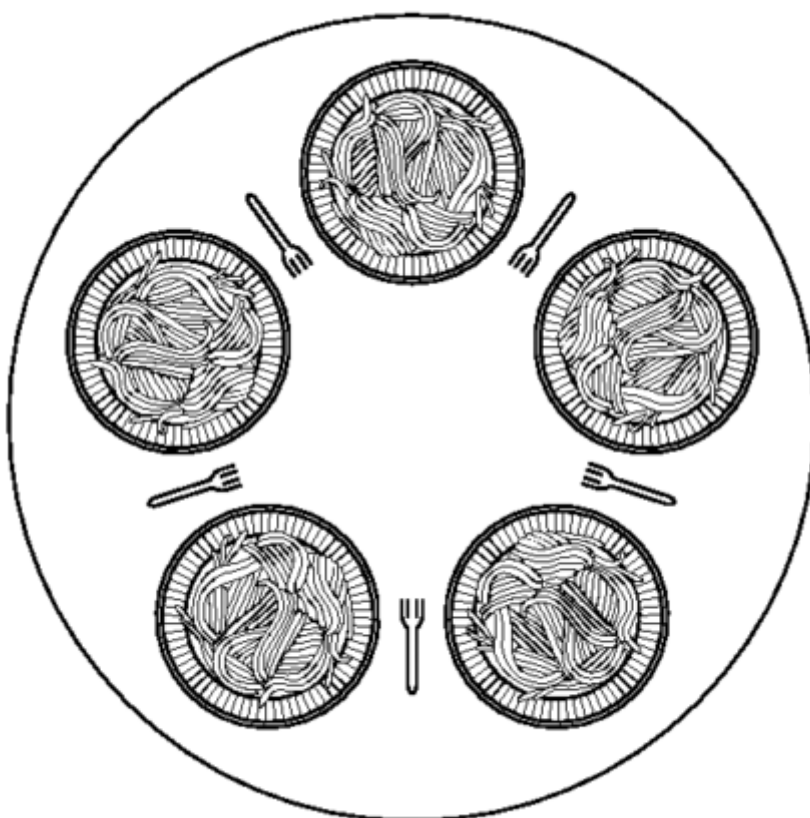


## Dining philosophers

Dining philosophers (De spisende filosoffer) er en sand klassiker, som Dijkstra konstruerede i 1965. Det er samtidig et eksempel på en finurlig anvendelse af semaforer.

Situationen er den, at en gruppe af filosoffer er samlet omkring middagsbordet. Der er spaghetti på menu'en. Enhver spaghetti entusiast ved at man til spisning af spaghetti kun behøver en gaffel, men af ukendte årsager skal filosoferne i vores eksempel bruge to gaffler til at spise spaghetti. Desværre er det lidt små med gaffler - faktisk har man kun, hvad der svarer til én gaffel pr. filosof. Man har placeret gafflerne på det obligatoriske runde bord, således at der er en gaffel imellem hver tallerken:



Hver filosof skal nu bemægtige sig de to gaffler ved hans tallerken. Det er naturligvis en fremgangsmåde der giver nogle forviklinger - hvilket er det interessante ved problemet.

Filosoffer er jo civiliserede mennesker og når de har haft lejlighed til at spise lidt spaghetti lægger de igen begge gaffler, så deres kolleger også kan få lejlighed til at spise - og så fremdeles.

## Ineffektiv løsning

Det er åbenlyst at vi har et samtidighedsproblem, og det er meget enkelt at løse. Vi sætter blot en mutex lås på det hele og lader en filosof spise af gangen. Løsningen er nem nok, men den er ikke optimal. **Med flere gaffler burde det kunne lade sig gøre at flere filosoffer kunne spise på en gang.** På figuren ovenfor burde der f.eks. kunne være to filosoffer, der spiste samtidig

## Deadlock

Desværre dur det ikke at lade filosofferne selv tage en gafflerne, for derved kan der opstå deadlock. Hvis alle filosoffer samtidig tager den venstre (eller den højre) gaffel, sidder de alle med en gaffel, og så sker der ikke mere!

Idéen i den løsning vi her skal se, er at vi sætter en mutex/monitor lås på det at undersøger om man kan tage de to gaffler man skal bruge. En filosof der ønsker at spise vil derfor først skulle tage mutex/monitor låsen. Dernæst undersøger han om han kan tage de to gaffler. Hvis det er muligt, tager han dem og frigiver låsen. Er det derimod ikke muligt sætter han sig til at vente.

Se eventuelt ProducerConsumer Simulation på Praxis.

## Video

<https://www.youtube.com/watch?v=ruovgwXyYs>

## Yderligere info

[https://en.wikipedia.org/wiki/Dining\\_philosophers\\_problem#Problem\\_statement](https://en.wikipedia.org/wiki/Dining_philosophers_problem#Problem_statement)

Du skal udvikle en konsolapplikation der simulerer Dining philosophers.

For at give alle filosofferne en fair chance for at overleve skal du implementere hver filosof som en tråd. Benyt Sleep og Random klasserne til at repræsenterer længden at tid en filosof tænker og spiser. Implementer eventuelt gafflerne som et Boolean array med en længde på 5, hvis en gaffel benyttes `fork[i] == true`, eller falsk.

Hint: overvåg fork array i en Monitor kan redde filosofferne fra en tidlig død

Output:

```
file:///c:/users/administrator/documents/visual studio 2015/Projects/Dining phil...
Philosopher 0 is eating...
Philosopher 1 is eating...
Philosopher 4 is eating...
Philosopher 3 is eating...
Philosopher 2 is eating...
Philosopher 1 is eating...
Philosopher 0 is eating...
Philosopher 4 is eating...
Philosopher 1 is eating...
Philosopher 2 is eating...
Philosopher 0 is eating...
Philosopher 4 is eating...
Philosopher 1 is eating...
Philosopher 2 is eating...
Philosopher 3 is eating...
Philosopher 4 is eating...
Philosopher 0 is eating...
Philosopher 1 is eating...
```

Udvid dit program så man kan se om en filosof spiser, tænker eller venter.