

Introduktion til Programmering og Problemløsning (PoP)

Jon Sparring
Department of Computer Science
2020/09/23

UNIVERSITY OF COPENHAGEN



Corona – guidelines



Velkommen til hold-/auditorieundervisning

Vi skal følge følgende retningslinjer:



- Mød ikke op, hvis du har symptomer på COVID-19 – også lette
- Host/nys i ærmet



- Hold afstand på minimum 1 meter,
- Undgå at bruge elevatorer. Hold 2 m afstand, hvis de bruges.



- Sprit hænder, når du går ind i bygningen
- Sprit hænder når du går ind i UV-lokalet
- Sørg for god håndhygiejne - grundigt og ofte



- Ved indgang i lokalet; sæt jer, så tæt passage af andre minimeres
- Vær opmærksom på, hvor I må/ikke må sidde
- Sid med min. 1 m afstand
- Undgå at gå rundt under undervisningen



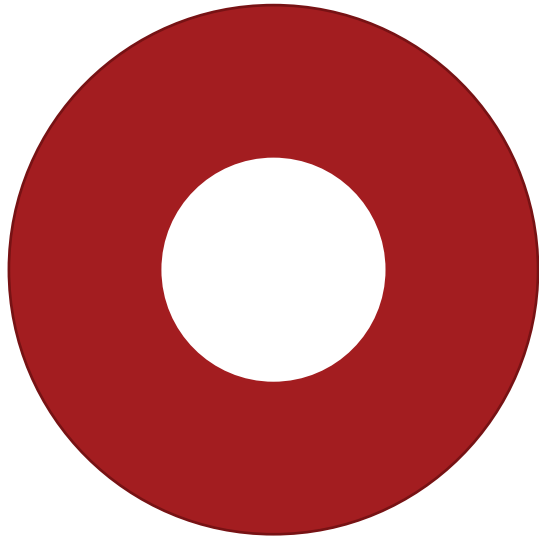
- Meld dig til at rengøre pladser efter undervisning – husk at rengøre sprayflaske efter brug
- Del ikke udstyr med andre



- Tøm lokalet tættest på udgangen først
- Forlad lokalet i god ro og orden
- Hold afstand til hinanden



Kodegenbrug



Hvad er arealet af en annulusen med ydre og indre diameter på 7 cm og 3 cm?

Opgavededdeling:

1. Areal af en cirkel
2. Areal af en annulus

```
let areaCircle r =  
    printfn "%g" (System.Math.PI * r * r)  
let areaAnnulus R r =  
    printfn "%g" (areaCircle R - areaCircle r)
```

```
let areaCircle r =  
    System.Math.PI * r * r  
let areaAnnulus R r =  
    areaCircle R - areaCircle r  
  
printfn "%g" (areaAnnulus 3.5 1.5)
```

Virkefelter vs. Miljø (scope vs. environment)

Virkefelt

```
let areaCircle r =
```

```
  System.Math.PI * r * r
```

```
let areaAnnulus R r =
```

```
  areaCircle R - areaCircle r
```

```
printfn "%g" (areaAnnulus 3.5 1.5)
```

areaCircle-r's
virkefelt

areaCircles
virkefelt

areaAnnulus-R
og -r's
virkefelt

areaAnnulus'
virkefelt

Miljø

```
let areaCircle r =
```

```
  System.Math.PI * r * r
```

```
let areaAnnulus R r =
```

```
  areaCircle R - areaCircle r
```

```
printfn "%g" (areaAnnulus 3.5 1.5)
```

areaCircle-krop
kender r (men ikke
areaCircle)

areaAnnulus-
krop kender R, r
og areaCircle

printfn kender
areaCircle og
areaAnnulus

Håndkøring

areaCircle-krop
kender r (men ikke
areaCircle)

areaAnnulus-
krop kender R, r
og areaCircle

Miljø

```

1 let areaCircle r =
2   System.Math.PI * r * r
3
4 let areaAnnulus R r =
5   areaCircle R - areaCircle r
6
7 printfn "%g" (areaAnnulus 3.5 1.5)
```

printfn kender
areaCircle og
areaAnnulus

Step	Linje	Miljø	Binding
1	1	E ₀	areaCircle = ((r),krop,())
2	4	E ₀	areaAnnulus = ((R,r),krop,(areaCircle))
3	7	E ₀	areaAnnulus 3.5 1.5 = ?
4	4	E ₁	((R=3.5,r=1.5),krop,(areaCircle))
5	5	E ₁	areaCircle 3.5 = ?
6	1	E ₂	((r=3.5),krop,())
7	2	E ₂	Return = 38.4 ...
8	5	E ₁	areaCircle 3.5 = 38.4...
9	5	E ₁	areaCircle 1.5 = ?
10	1	E ₃	((r=1.5),krop,())
11	2	E ₃	Return = 7.0 ...
12	5	E ₁	areaCircle 1.5 = 7.0...
13	5	E ₁	return = 31.4...
14	7	E ₀	areaAnnulus 3.5 1.5 = 31.4...
15	7	E ₀	Output = 31.4...
16	7	E ₀	Return = ()

Advarsel: biblioteks- og applicationsfil skal have forskellige navne!

area.fs

module area

let areaCircle r = System.Math.PI * r * r

let areaAnnulus R r = areaCircle R - areaCircle r

~~area.fsx~~

~~printfn "%g" (area.areaAnnulus 3.5 1.5)~~

areaTest.fsx

printfn "%g" (area.areaAnnulus 3.5 1.5)

Spørgetime

- Hvornår bruger man val og type i signaturfilen og hvad er forskellen?
 - Med "val" forventes en binding. "type" er en type som "float"
- Hvornår er det ok at bruge globale variabler i en funktion?
 - Næsten aldrig!
- Hvad er forskellen på at compile en .fsx fil og en .fs fil til et bibliotek?
 - Meget lidt, se kapitel 12 i <https://fsharp.org/specs/language-spec/4.1/FSharpSpec-4.1-latest.pdf>