

## COS lektion 9, den 19. november 2021.

### Praktisk information

Planen for resten af COS kurset:

Lektion 9, den 19. november 2021.

kl.: 10:15-12:00: Algoritmer.  
kl.: 12:15-14:00: Instruktortimer. Øvelser.

Lektion 10, den 26. november 2021.

Software Engineering:  
kl.: 10:15-11:00: Algoritmer fortsat. Gruppedannelse (3 studerende pr. gruppe).  
kl.: 12:15-14:00: Instruktortimer. Miniprojekt.

Software Teknologi:  
11:15 – 12:00: Intro til CPS/IoT.  
Gruppedannelse (3 studerende pr. gruppe)  
kl.: 12:15-14:00: Instruktortimer.  
Installation af software,  
Miniprojekt.

Lektion 11, den 3. december 2021.

Software Engineering:  
kl.: 10:15-11:00: Workshop som er et lille algoritme opgave udført af grupperne,  
kl.: 12:15-14:00: Instruktortimer. Miniprojekt.

Software Teknologi:  
kl. 11:15 – 12:00: Workshop som er et lille simulationsprojekt udført af grupperne.  
kl.: 12:15-14:00: Instruktortimer. Miniprojekt.

Lektion 12, den 10. december 2021.

kl.: 10:15-12:00: Opsamling og eksamensforberedelse.  
kl.: 12:15-13:00: PA2 dækkende stoffet fra lektion 4 – 11.  
kl.: 13:15-14:00: Afslutning og de sidste råd fra instruktorerne.

## Emner til lektion 9

Algoritmer.

## Forberedelse til lektion 9

Gennemlæs afsnittene 5.1, 5.2 og 5.3.

Løs alle *Questions & Exercises* (dem til 5.3 er lidt komplicerede; men prøv at løse dem før i kigger i facitlisten).

Prøv at implementere nogle af problemerne i Python.

## Bearbejdelse af dagens emner

Afsnit 5.1 snakker vi lidt kort om på baggrund.

Afsnit 5.2: Vi snakker om hvordan en algoritme kan repræsenteres.

Vi kigger på pseudokode for *Iterativ Euclid* og implementerer algoritmen i Python.

Afsnit 5.3 vi snakker om hvordan man kan finde algoritmer.

## Opgaver til løsning i instruktortimerne 12:15 – 14:00

*Chapter Review Problems 1, 2, 3, 4, 5 og 6 på side 310 (263).*

*Chapter Review Problem 18 på side 311 (264).*

Og så er der lige 2 opgaver som skal implementeres i Python:

### **Leap year**

The normal year contains 365 days, but the leap year contains 366 days.

Logically, All the years that are perfectly divisible by 4 are called as Leap years except the century years.

Century year's means they end with 00 such as 1200, 1300, 2400, 2500 etc. (Obviously they are divisible by 100). For these century years we must calculate further to check the Leap year.

- If the century year is divisible by 400 then that year is a Leap year
- If the century year is not divisible by 400 then that year is not a Leap year

### **Easter Sunday**

Easter Sunday is the first Sunday after the first full moon of spring. To compute the date, you can use this algorithm, invented by the mathematician Carl Friedrich Gauss in 1800:

1. Let  $y$  be the year (such as 1800 or 2001).
2. Divide  $y$  by 19 and call the remainder  $a$ . Ignore the quotient.
3. Divide  $y$  by 100 to get a quotient  $b$  and a remainder  $c$ .
4. Divide  $b$  by 4 to get a quotient  $d$  and a remainder  $e$ .
5. Divide  $(8 * b + 13)$  by 25 to get a quotient  $g$ . Ignore the remainder.
6. Divide  $19 * a + b - d - g + 15$  by 30 to get a remainder  $h$ . Ignore the quotient.
7. Divide  $c$  by 4 to get a quotient  $j$  and a remainder  $k$ .
8. Divide  $a + 11 * h$  by 319 to get a quotient  $m$ . Ignore the remainder.
9. Divide  $2 * e + 2 * j - k - h + m + 32$  by 7 to get a remainder  $r$ . Ignore the quotient.
10. Divide  $h - m + r + 90$  by 25 to get a quotient  $n$ . Ignore the remainder.
11. Divide  $h - m + r + n + 19$  by 32 to get a remainder  $p$ . Ignore the quotient.

Then Easter falls on day  $p$  of month  $n$ .

For example, if  $y$  is 2001:  $a = 6$ ,  $b = 20$ ,  $c = 1$ ,  $d = 5$ ,  $e = 0$ ,  $g = 6$ ,  $h = 18$ ,  $n = 4$ ,  $j = 0$ ,  $k = 1$ ,  $p = 15$ ,  $m = 0$ ,  $r = 6$ . Therefore, in 2001, Easter Sunday fell on April 15.

Write a program that prompts the user for a year and prints out the month and day of Easter Sunday.

Test:

- 15. april 1900
- 30. marts 1997
- 16. april 2017
- 1. april 2018
- 21. april 2019

## Emner til næste lektion (Lektion 10 den 26. november)

### **Lektion 10se:**

Kap 5.4:

Studer eksemplet The Sequential Search Algoritm, fig 5.6 og løs Q&E 1 på side 289 (244).  
Implementer og test løsningen i Python.

Gennemlæs afsnittet om Loop Control og løs Q&E 2 på side 289 (244).

Studér Insertion Sort Algoritmen meget grundigt, samt Q&E 4 og 5 på side 289 (244).  
Implementer og test algoritmen i python.

Kap 5.5 går vi ikke helt dybt ind i; men prøv at forstå principperne og løs de 4 Q&E på side 299 (253).  
Kap 5.6 skal blot skimmes igennem.

Udlevering af algoritme opgave

### **Lektion 10st:**

Orienter jer i følgende materialer, så i får en fornemmelse for hvad CPS og IoT er for noget:

- [https://en.wikipedia.org/wiki/Internet\\_of\\_things](https://en.wikipedia.org/wiki/Internet_of_things)
- [https://en.wikipedia.org/wiki/Cyber-physical\\_system](https://en.wikipedia.org/wiki/Cyber-physical_system)
- Lagt på **itslearning/resources/ ST: Materialer til CPS/IOT Lektion 10 og 11:**
  - IoT for Dummies
  - 2018 - LNCPS - Chapter 1. What is a Cyber Physical System\_.pdf
  - Leverage\_Intro\_to\_IoT\_eBook.pdf

Udlevering af simulationsprojekt.