

# Obligatorisk oppgave 4

Due 1 Nov by 23:59      Points 1      Submitting a website url  
Available 10 Oct at 0:00 - 15 Nov at 23:59 about 1 month

## Oblig 4 - Infprog

Denne obligen har 3 ulike alternativ:

Alternativ A: For de som har lyst til å gjøre et eget prosjekt

Alternativ B: For de som vil ha en gitt prosjektoppgave

Alternativ C: For de som ønsker "drille" basisferdigheter - Et utplukk av gamle eksamensoppgaver

Gjøres alternativ A eller B må omfanget som et minimum tilsvare omfanget av alternativ C. Selvsagt kan dere også gjøre ett alternativ fullt ut og fortsette på et annet.

### Alternativ A:

Lag ditt egendefinerte prosjekt. Veldig gjerne et spill, men prøv å unngå et spill som er "for grafisk", da det i så fall er logikken bak som er beste trening til eksamen.

Sørg for at dere får drillet bruk av løkker, arrayer og funksjoner.

Pass også på å ikke gå for høyt ut. Selv spill som virker enkle, kan ta mye tid å lage. Det er bedre heller å legge til funksjoner eller "rydde i kode" dersom dere får god tid.

### Alternativ B:

Du skal lage et hangman-spill ved å benytte canvas.

MERK: En del av oppgaven er å selv utforske hvordan man tegner ulike grafiske elementer på et canvas. Dette har altså ikke vært spesielt nøye gjennomgått i timene og er en øvelse i å lese andres eksempler/manualer. Husk at dere skal lese, forstå og lage selv. Ikke kopiere... IKKE søk opp eksempler på hangman for å få innspill til hvordan det gjøres. Da misser dere alt av trening i problemløsning, noe dere må kunne på eksamen...

Forsøk å bygge opp spillet ved hjelp av funksjoner. En funksjon bør som en tommelfingerregel ikke ha mer enn 20-30 linjer kode som maks.

Følgende funksjonalitet er et minimum (det er lurt å løse de to første punktene som hvert sitt lille prosjekt, og så sette de sammen i punkt 3...):

- For hver gang man trykker med musen på canvas-elementet skal neste del av hangman vises.
- Spilleren skal kunne skrive inn en bokstav og få en eller annen form på feedback på om bokstaven er med i ordet eller ikke.
- Sett sammen de to delene, slik at man får en minimumsutgave av "hangman"
- Når siste del av hangman vises skal det stå "game over" i canvas-elementet. Klarer vi ordet skal vi vise en gratulasjon

Følgende funksjonalitet må med for å løse oppgaven fullt ut:

- Om bokstaven finnes i ordet skal denne/disse bokstavene bli synlige i løsningsordet der bokstaver ellers er kamuflert som \_
- Om bokstaven ikke finnes skal neste del av hangman vises.
- Løsningsord skal velges blant en liste forhåndsdefinerte ord.

Følgende funksjonalitet hadde vært artig (for de som vil gjøre noe ekstra):

- Vanskelighetsgrad som for eksempel
  - Basert på antall bokstaver i ordet
  - Basert på tid(countdown)
  - Eller andre ting du selv kommer på
- Poengsum som viser hvor godt spilleren løste oppgaven. Dette kan f.eks være
  - Tid brukt
  - Forsøk brukt
  - Vanskelighetsgrad
- Programmeringsanimert konfetti på gratulasjonsmeldingen er stas <= Dette kan være mye jobb, så ta det som en utfordring de som vil. HINT: Legg en ny canvas ovenpå den andre ved hjelp av (z-index) og tegn konfetti på denne.
- Andre ting du selv kommer på

## Alternativ C

*Det er ikke gitt at oppgavene står i riktig rekkefølge mhp vanskelighetsgrad. Omfanget tilsvarer ca en eksamensoppgave vil jeg gjette på, men litt for mange oppgave av "samme type" i forhold til en eksamen..*

Vanskelighetsgraden er vel også ca lik, men det mangler en "nøtt" for de som vil ha B/A. ettersom det er gamle eksamensoppgaver er det ikke alle oppgavene som etterspør all koden som trengs " for å kjøre". Legg på dette selv for å teste.

Du skal gjøre som et minimum gjøre 6 av de 9 oppgavene. Hvilke velger du selv.

I en eksamenssetting burde alle 9 oppgavene da tatt ca 4 timer, om man skal ha en A.

## Oppgave 1

Skriv kode som produserer følgende utskrift (teksten "..." skal selvsagt erstattes med de virkelige utskriftene fra 8 til 99).

0\*0=0

1\*1=1

2\*2=4

3\*3=9

4\*4=16

5\*5=25

6\*6=36

7\*7=49

...

100\*100=10000

## Oppgave 2

Skriv ferdig følgende funksjoner.

1. a) Funksjon som returnerer absoluttverdien av et tall (Absoluttverdien er selve tallverdien uten fortegn, slik at 7 og -7 begge returnerer 7). Du får her ikke benytte deg av den innebygde abs-funksjonen.

function absoluttverdi(tall)

1. b) Funksjon som finner gjennomsnittet av elementene mellom en nedre og øvre index i en array.

```
function finnGjennomsnittUtvalg(array, startindex, sluttindex)
```

## Oppgave 3

Skriv ferdig følgende funksjoner:

1. a) En funksjon som returnerer om et tall er partall eller ei

```
function partall(tall)
```

1. b) En funksjon som returnerer summen av en array

```
function sum(liste)
```

1. c) En funksjon som returnerer om to arrayer (parameterne) har samme sum (Tips: Benytt funksjonen fra deloppgave b)

```
function sammeSum(listeA,listeB)
```

## Oppgave 4

Gå ut i fra at du har følgende variabler definert (verdiene er eksempelverdier) som forteller om ulike sider ved været for foregående time (gjennomsnittsverdier).

```
var temperatur = 12.3; // i °C
```

```
var vindhastighet = 4.6; // i m/s
```

```
var vindretning = "NØ" // forkortelse for NordØstlig
```

```
var nedbor = 3.7 // i antall mm
```

Skriv logiske uttrykk for følgende påstander:

1. a) Det regner og er vindstille
2. b) Det blåser nord-, nordøst- eller nordvestlig vind
3. c) Det regner mer enn 5 mm ELLER det er frost og sørøstlig vind

## Oppgave 5

## Oppgave 6

5/9

## Oppgave 7

Du har følgende tre assosiative arrayer, som inneholder en kobling mellom alle bokstaver i alfabetet og et ord. Under vises kun A til D for hver array for å spare plass:

```
var arr1 = {A: "gale", B: "snartenkte", C: "stinkende", D: "rykende" };
```

```
var arr2 = {A: "buss", B: "nese", C: "glass", D: "tomat"};
```

```
var arr3 = {A: "muffins", B: "tastatur", C: "kopp", D: "vase"};
```

Din oppgave blir nå å lage en "kallenavngenerator", der brukeren skal taste inn sitt virkelige navn i et skjema som ser slik ut:

# Kallenavngenerator:

Fornavn:

Etternavn:

Systemet skal så plukke ut:

Første bokstav av fornavn: B

Første bokstav av etternavn: D

Siste bokstav av etternavn: A

Disse tre bokstavene skal så slås opp i hver sin array, for å hente ut de tre delene kallenavnet skal bestå av:

B i arr1 gir "snartenkte"

D i arr2 gir "tomat"

A i arr3 gir "muffins"

Delen fra arr1 skal gi det nye fornavnet. De to delene fra arr2 og arr3 skal settes sammen og gi det nye etternavnet:

Fornavn: "Snartenkte"

Etternavn: "Tomatmuffins"

Til slutt skal det skrives ut en melding til brukeren:

Velkommen Bjarne Dunna!

Du er herved døpt *Snartenkte Tomatmuffins*.

## Oppgave 8

Du har to arrayer med tall (kalt *a* og *b*) som er like lange. f.eks:

```
var a = [2, 5, 2, 4, 7, 3, 2];
```

```
var b = [9, 1, 3, 2, 6, 4, 3];
```

Du har også en array (kalt *c*) med tilsvarende lengde, der alle verdier er 0:

```
var c = [0, 0, 0, 0, 0, 0, 0];
```

Skriv kode som gjør at array *c* får det største tallet fra *a* og *b* på hver posisjon/indeks. Med eksempeldataene vist over vil altså arrayen *c* få innholdet:

```
[9, 5, 3, 4, 7, 4, 3];
```

(Denne oppgaven krever altså ingen utskrift. Kun at arrayen fylles.)

## Oppgave 9

Ta utgangspunkt i at du har en array med alle bokstaver i det engelske alfabetet:

```
var alfabet = ["A","B","C","D","E", osv, "Z"];
```

Du har også en array som inneholder et utvalg bokstaver. Denne kan f.eks se slik ut:

```
var liste = ["F","B","C","F", osv, "P"];
```

Lag programkoden som skriver ut alle bokstavene i alfabetet som IKKE forekommer i arrayen liste.

TIPS: Gå gjennom arrayen alfabet. For hver bokstav, gå gjennom arrayen liste og se om bokstaven finnes der.

Some Rubric (1)		
Criteria	Ratings	Pts
Kode <a href="#">view longer description</a>		0.0 pts
Funksjonalitet <a href="#">view longer description</a>		0.0 pts
Brukergrensesnitt <a href="#">view longer description</a>		0.0 pts
Kreativitet <a href="#">view longer description</a>		0.0 pts
Bugs <a href="#">view longer description</a>		0.0 pts
Totalbedømming <a href="#">view longer description</a>		100.0 pts
		Total points: 100.0



