

# Proefwerk Javascript Processing havo versie C

Dit proefwerk bestaat uit een theoretisch deel dat je op papier maakt en een praktisch deel dat je op de computer maakt. In totaal heb je 60 minuten de tijd. Je mag pas beginnen met het praktische deel als je jouw uitwerkingen van het theoretische deel hebt ingeleverd. Het is je eigen verantwoordelijkheid om de tijd goed te verdelen!

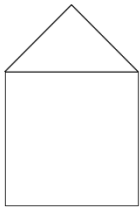
Het praktische deel maak je op de computer met behulp van het programma Notepad++. Hiermee hebben we in de les geoefend. Elke keer als jij vindt dat je een opdracht af hebt, laat je het resultaat zien aan de surveillant. Deze zal dat onderdeel dan beoordelen.

Zit je vast? Je mag vragen om een hint, maar dat kost je wel punten op je beoordeling!

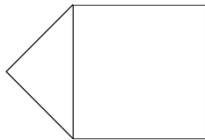
Is het tijd en heb je een opdracht nog niet af? Of wil je verder met een andere opdracht omdat je niet verder komt? Laat je werk dan wel beoordelen! Het kan zijn dat je wel een deel van de punten hebt behaald!

## Theoretische deel (TOTAAL 6 • 5 = 30 punten)

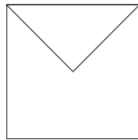
1. Bekijk de code in het blok hiernaast. Wat is het resultaat wanneer deze code wordt uitgevoerd? A, B, C of D?



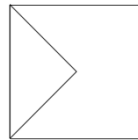
A



B



C



D

```
rect(0,0,200,200);  
triangle(0,0,0,200,100,100);
```

2. Met de code in het blok met bij vraag 1 maak je één van de getoonde figuren. We willen deze figuur tekenen met een functie *tekenFiguur*. Schrijf op met welke coderegels je deze functie kunt maken.

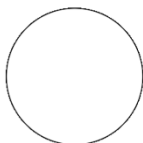
3. Bekijk de code in het blok hiernaast. Wat is het resultaat wanneer deze code wordt uitgevoerd? A, B, C of D?



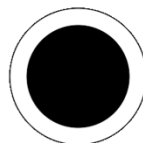
A



B



C



D

```
fill('black');  
ellipse(0,0,150);  
fill('white');  
ellipse(0,0,200);
```

4. Bekijk de code in het blok hiernaast.  
Hoe vaak wordt de functie *tekenHuis()* uitgevoerd?  
A: 4x B: 5x C: 6x

```
for (var a=0; a<=5; a++) {  
  tekenHuis();  
  translate(200,0);  
}
```

5. Bekijk de code in het blok hiernaast.  
Wat is de uiteindelijke waarde van de variabele *uitkomst*?  
A: 0 B: 5 C: 6 D: 7 E: 10 F: 15

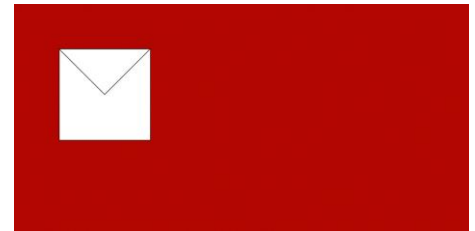
```
var uitkomst=0  
for (var a=0; a<=5; a++) {  
  if (uitkomst<a) {  
    uitkomst+=a;  
  }  
}
```

6. Leg in een Nederlandse zin uit wat er gebeurt als de coderegel **x=constrain(mouseX,0,100)** wordt uitgevoerd.

# Proefwerk Javascript Processing havo versie C

## Praktisch deel 1<sup>e</sup> opdracht (TOTAAL 30 punten)

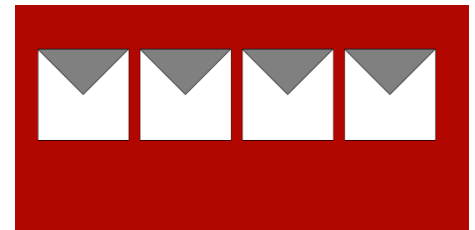
Open het bestand *h4vcpd1.html* en bekijk de code. Met het programma wordt een enveloppe getekend door de functie *tekenEnv* aan te roepen. Het resultaat zie je hiernaast in figuur 1.



FIGUUR 1

Hieronder staan een aantal opdrachten. Voer deze opdrachten uit (sla eventueel een stap over als je die niet weet):

- A. (10 punten) Geef de driehoek van de enveloppe een grijze (Engels: *grey*) kleur.
- B. (10 punten) Gebruik een herhaling om de functie *tekenEnv* vier keer aan te roepen.
- C. (10 punten) In regel 8 is een constante *AFSTAND* gedeclareerd. Voeg een regel toe aan de herhaling die er voor zorgt dat de vier enveloppes op een onderlinge afstand van 25 komen te staan, door gebruik te maken van deze constante. Jouw resultaat moet nu overeenkomen met figuur 2.

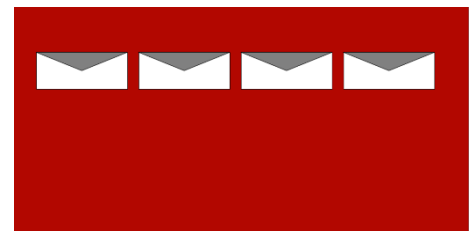


FIGUUR 2

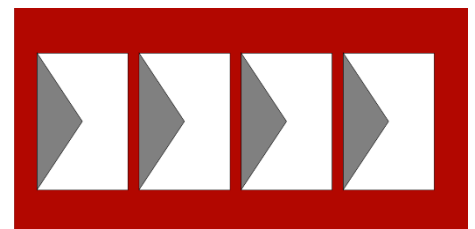
## Praktisch deel 2<sup>e</sup> opdracht (TOTAAL 30 punten)

Bij deze opdracht werk je door in het bestand *h4vcpd1.html*.

- D. (15 punten) De enveloppe heeft nu vaste afmetingen. We willen dat de hoogte van de enveloppe kan worden aangepast. De punt van de driehoek van de enveloppe moet bij elke hoogte in het midden van de figuur blijven. Pas de functie *tekenEnv* aan zodanig dat de regel *tekenEnv(80)* leidt tot het resultaat dat hiernaast staat getekend in figuur 3.
- E. (15 punten) De grijze driehoek van de enveloppe moet altijd aan de lange zijde zitten. De enveloppes hebben een vaste breedte van 200. Als de opdracht *tekenEnv(300)* wordt gegeven, moet de tekening er daarom uitzien zoals in figuur 4. Zorg dat de grijze driehoek aan de linkerkant wordt getekend (zoals getoond in figuur 4), als de hoogte van de enveloppe groter is dan 200.



FIGUUR 3



FIGUUR 4

EINDE