

Oplossing 1: 2 willekeurige getallen tussen de 2 en 12 genereren.

Het eerste gedeelte had je vast wel. De namen van de variabelen mogen natuurlijk anders zijn dan hieronder.



Nu moet jij met deze getallen de vraag samenstellen.

Daarvoor moet de **“voeg - en - samen”** functie worden gebruikt. Hiermee kunnen teksten en getallen worden samengevoegd. Wanneer je meer dan 2 teksten wilt samenvoegen, kan je deze opdracht nesten (=in elkaar herhalen). De samengestelde teksten moeten in een variabele (bijvoorbeeld; **“De vraag”**) worden gezet.

Hieronder 3 voorbeelden. **Let op:** Zijn voorbeelden, niet precies zo overnemen!

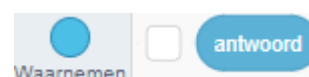


De vraag moeten we aan de speler stellen en wachten op het antwoord.

Oplossing 2: Sprite **Gobin** vervolgens de speler de opgave met deze getallen laat zien en vraagt naar het antwoord



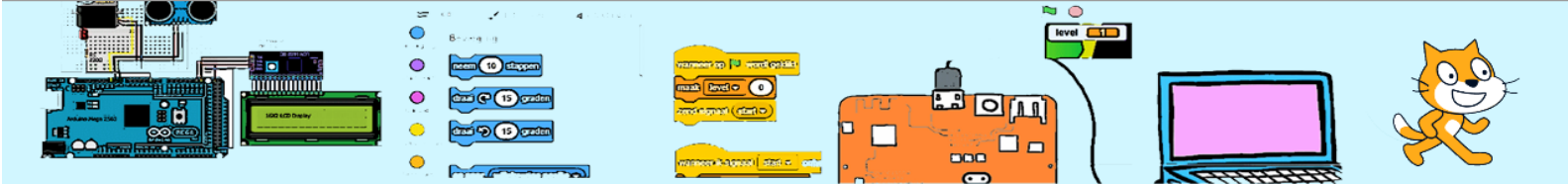
Na een vraag bewaart Scratch het antwoord van de speler in de variabele **“antwoord”**. Deze variabele is te zien in de lichtblauwe groep **“Waarnemen”**.



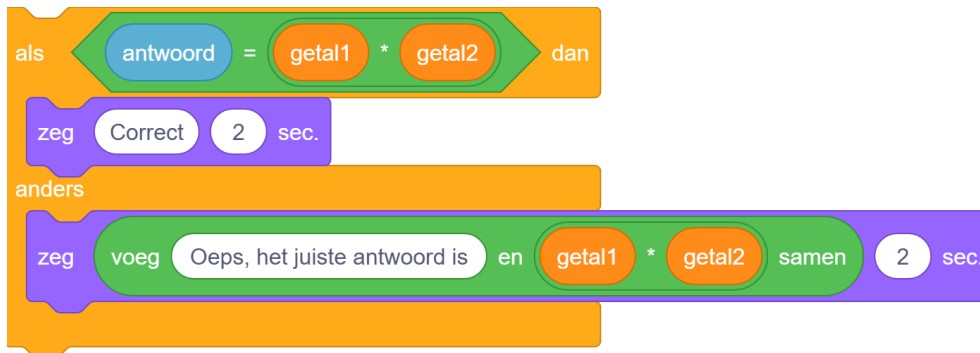
Opdracht 3:

Het antwoord van de speler controleren en de speler laten weten of het gegeven antwoord goed of fout is.

=> **Probeer het eerst zelf voordat je hieronder kijkt naar de oplossing.**



Oplossing 3:

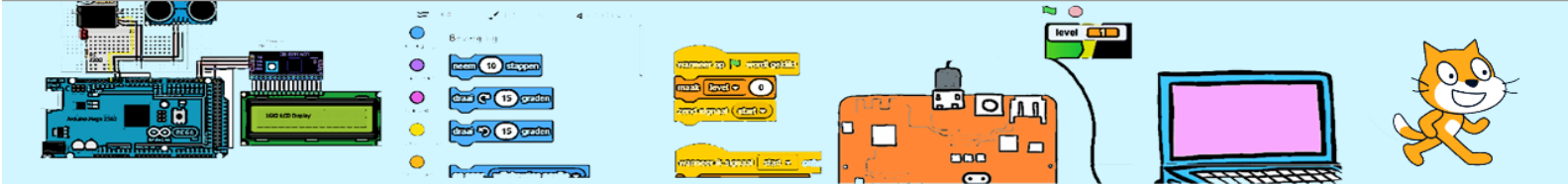


Probeer de code uit. Werkt het?

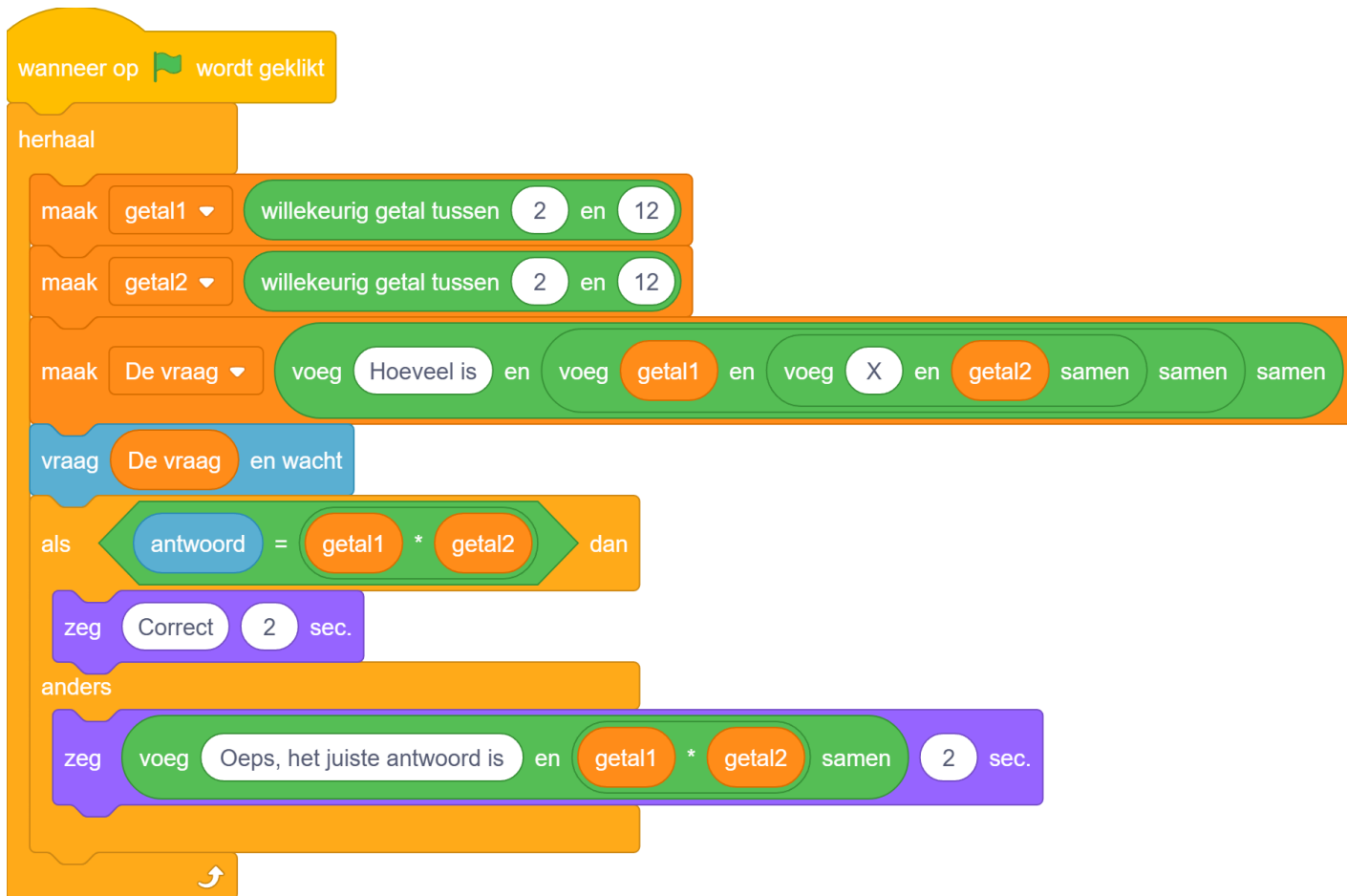
3. Meerdere sommen na elkaar

Opdracht 4: Laten we meerdere rekensommen na elkaar maken.

=> Probeer weer eerst zelf een oplossing te vinden alvorens je hieronder kijkt.



Oplossing 4: Laten we meerdere rekensommen achter elkaar vragen.



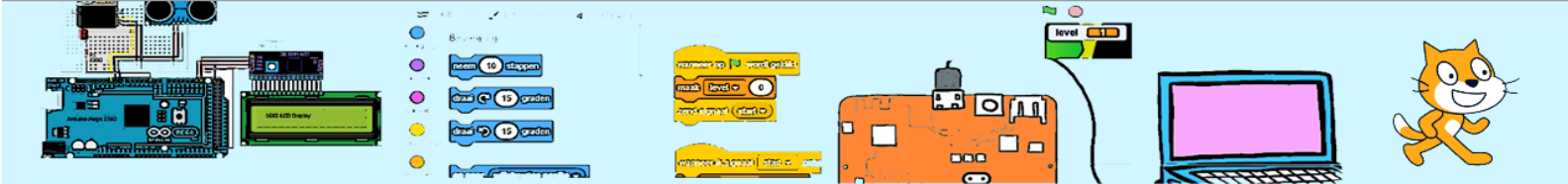
4. Een tijd klok toevoegen

Opdracht 5: Laten we een tijdslimiet toevoegen van bijvoorbeeld 30 seconden.

Hier hebben we een klok voor nodig. Hoe zouden we een klok kunnen maken?

Hint: We beschikken over een "wacht 1 sec"-opdracht.

=> Probeer weer eerst zelf een oplossing te vinden alvorens je hieronder kijkt.



Oplossing 5: Laten we een tijdslimiet toevoegen van bijvoorbeeld 30 seconden.

We maken eerst een variabele “tijd”. We wachten steeds 1 seconde en dan hogen we de variabele met 1 op alvorens we opnieuw 1 seconde wachten.

Dus we tellen het aantal seconden.

De variabele “tijd” moet op 0 worden gezet wanneer de quiz start. We laten de tijd zien op het speelveld door de checkbox voor de variabele “tijd” aan te vinken.



Na 30 seconden verzenden we een signaal “tijdsOp” aan alle scripts (blokken code) die hier naar luisteren.

Nu weten we wanneer de tijd op is. Maar hoe gaan we dit gebruiken?

We maken een stukje code dat wacht op het bericht “tijdsOp” en dan direct alle andere scripts stopt. Dit script zelf stopt niet. En kan bijvoorbeeld een boodschap op het scherm zetten.



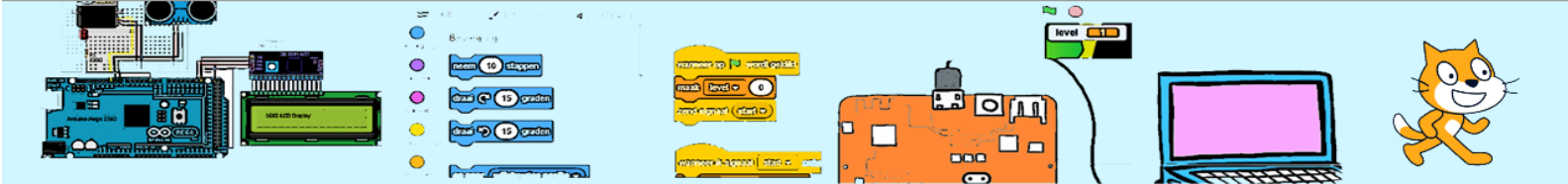
5. Aantal goed

We zijn aardig op weg.

Opdracht 6: We willen nog een score toevoegen van het aantal goed beantwoorde sommen.

Hiervoor moeten we een variabele “Aantal goed” toevoegen. Deze variabele laten we net als de tijd ook op het scherm zien.

=> **Werk dit zelf uit voordat je naar de volgende paragraaf gaat.**



Oplossing 6: We willen nog een score toevoegen van het aantal goed beantwoorde sommen.

De variabele “Aantal goed” moet bij het starten van het spel op 0 worden gezet.



Bij ieder goed antwoord moet deze variabele met 1 worden verhoogd.



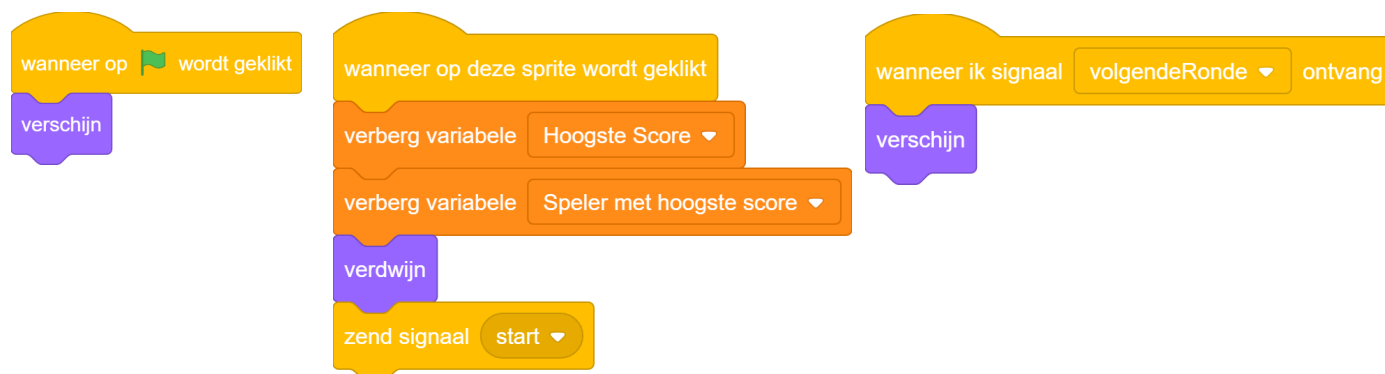
6. Hoogste score

Opdracht 7: We gaan het spel zo maken dat het meerdere keren achter elkaar kan worden gespeeld door meerdere (verschillende) personen.

En we gaan bijhouden wat de hoogste score is en wie de heeft behaald.

Om de spelers om de beurt te kunnen laten starten, maken we een “Start”-button. Bijvoorbeeld de sprite “**Button1**” waarin de tekst “Start” is geschreven. Wanneer erop geklikt wordt, wordt een signaal “start” verstuurd. De hoofd-lus van de sprite **Gobin** moet nu gaan starten wanneer het signaal “start” wordt ontvangen.

Hieronder staat de code voor de nieuwe “**Button1**”-sprite.

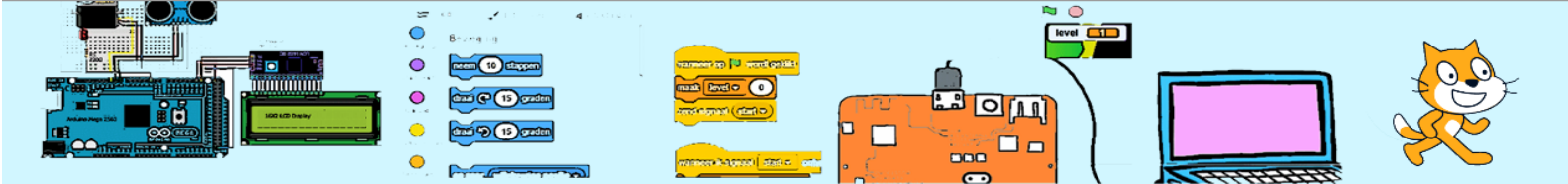


Wanneer het signaal “volgendeRonde” wordt ontvangen moet de knop weer verschijnen. Dit nieuwe bericht gaan we verzenden vanuit de sprite **Gobin**,

Bij **Gobin** moet nu de hoofd-lus worden gestart als het signaal “start” is ontvangen. Na het klikken op de groene vlag zorgen we ervoor dat alle variabelen geïnitieerd (van een beginwaarde voorzien) worden. Dit doen we omdat anders de inhoud willekeurig is. Er zou iets kunnen staan van een vorige run en of zelfs uit een ander Scratch programma.

Nu moeten we controleren of het **Aantal Goed** hoger is dan de **Hoogste Score** en als dat het geval is de **Hoogste Score** bijwerken, de naam van de speler vragen en deze bewaren. Ook moet de code voor de klok gaan starten na het klikken op de groene vlag. Check of je code werkt.

=> **Probeer het eerst zelf, kijk daarna pas naar een oplossing hieronder.**



Oplossing 7: Hoogste score en meerdere keren achter elkaar kunnen spelen.

```

wanneer op vlag wordt geklikt
maak Hoogste Score 0
maak Speler met hoogste score 0
  
```

```

wanneer ik signaal start ontvang
maak Aantal Goed 0
herhaal
  maak getal1 willekeurig getal tussen 2 en 12
  maak getal2 willekeurig getal tussen 2 en 12
  maak De vraag voeg Hoeveel is en voeg getal1 en voeg X en getal2 samen samen samen
  vraag De vraag en wacht
  als antwoord = getal1 * getal2 dan
    verander Aantal Goed met 1
    zeg Correct 1 sec.
  anders
    zeg voeg Oeps, het juiste antwoord is en getal1 * getal2 samen 2 sec.
  herhaal
  
```

```

wanneer ik signaal tijdIsOp ontvang
stop andere scripts in sprite
zeg Tijd is op! 2 sec.
als Aantal Goed > Hoogste Score dan
  zeg Gefeliciteerd. Je hebt de hoogste score tot nu toe 2 sec.
  vraag Hoe heet je? en wacht
  maak Speler met hoogste score antwoord
  maak Hoogste Score Aantal Goed
  toon variabele Speler met hoogste score
  toon variabele Hoogste Score
  zend signaal volgendeRonde en wacht
  
```

```

wanneer ik signaal start ontvang
maak tijd 0
herhaal tot tijd = 30
  wacht 1 sec.
  verander tijd met 1
  zend signaal tijdIsOp
  
```