

UNIVERSITEIT VAN AMSTERDAM 2018-2019

minor programmeren

Tentamen Programmeren 1

maandag 24 september 2018 9:00–10:30

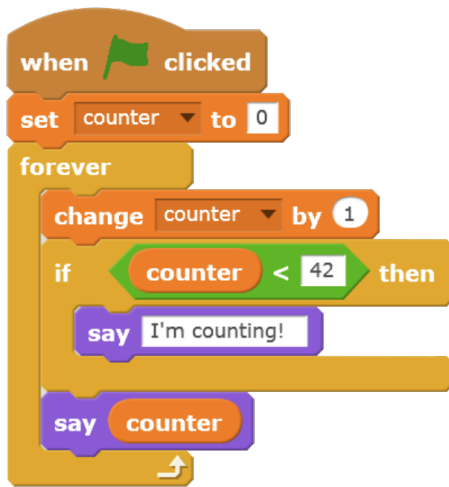
Vul hier je naam en studentnummer in vóór je begint: 	/ 37 p
---	--------

1. Je mag de vragen in Engels of Nederlands beantwoorden.
2. Dit is een "gesloten boek"-tentamen. Je mag voor het invullen je pen of potlood gebruiken, maar verder niets. Schrijf duidelijk en niet te groot.
3. Leg je studentenkaart (of ander ID met foto) klaar op je tafel. We komen langs om te kijken of je hierboven je naam hebt ingevuld en of deze klopt met je ID.
4. Laat het weten als je kladpapier nodig hebt.
5. Als je vragen hebt over hoe we iets bedoelen, dan kunnen we dat waarschijnlijk niet beantwoorden zonder een deel van het antwoord weg te geven (maar voel je vrij om het te proberen!).
6. Je hoeft geen comments in je code te schrijven.

Looping back to Scratch.

5 p **Question 1.**

Consider the Scratch script below.



In the space below, complete the translation of this Scratch script to a C program in such a way that its output is equivalent. (Your program's structure needn't be equivalent.) Assume that **counter** is an `int`, that **say** is `printf`, **set** means to assign a value to a variable, and **change** means to increment or decrement. Output `\n` after each line of text.

You may use any of the functions that you've learned in lectures or in the assignments. Pseudocode may earn you points, but usually less than correct C code!

```
#include <cs50.h>
#include <stdio.h>
```

```
int main(void)
{
```

Code reading and tracing.

2p Question 2.

Evaluate the following expressions:

a) $1.0 * 3.0 / 2$

b) $1 / 1 - 1.5$

c) $3 * 3 \% 6$

d) $3 + 4.0 / 2.0$

2p Question 3.

Evaluate the following expressions:

a) $0.5 \geq 0 \ \&\& \ 1.0 > 1 / 3.0$

b) $!(2 \geq 0.5) \ || \ 1.5 \geq 2$

c) $!(1.5 \geq 2) \ \&\& \ 1 \% 4 \geq 2.0$

d) $1.5 \geq 1.0 \ || \ 3.0 \leq 4 / 3$

2p Question 4.

Write down the final value of each variable after running the following programs:

a)

```
gee = 1 * 2
geo = gee + 1
pus = gee - 3
nan = gee + 1.5
pus = nan - 0.5
pus = geo / 1.5
```

b)

```
waf = 3 + 2
vow = waf - 1.0
nth = vow / 2
vow = waf + 1
vow = vow % 3
vow = vow + 3
```

2p Question 5.

Write down the final value of each variable after running the following programs:

a)

```
tyt = 3 / 2.0
taj = 2 * 2
if(taj <= 1.0)
    tyt = tyt - 3
if(tyt == 2.0)
    taj = taj % 5
else
    tyt = tyt * 2
```

b)

```
nod = 1.5 / 1.0
bar = 1.0 - 3
if(bar <= 1.5)
    bar = bar + 2
else if(bar <= 1.0)
    nod = nod / 1.5
else
    nod = nod + 1.5
```

2p **Question 6.**

Write down the final value of each variable after running the following programs:

a)

```
ofo = 8
uva = 10
while(ofo < 291 && uva > 3)
    ofo = ofo * 10
    uva = uva * 3
```

b)

```
bus = 45
ham = 2
while(bus > -7 || ham < 8)
    bus = bus - 8
    ham = ham * 2
```

3p **Question 7.**

Write down an algorithm that prints the first 256 values of the following sequence:

5 3 1 -1 -3 -5 -7 ...

3p **Question 8.**

Write down an algorithm that prints the first 12 values of the following sequence:

100000 10000 1000 100 10 ...

2p **Question 9.**

Write down what is printed when the following program is run:

```
s = "wahaamlloo"
for(int ann = 6; ann > 1; ann = ann - 2)
    print(s[ann / 2])
```

2p **Question 10.**

Write down what is printed when the following program is run:

```
s = "salurubadmux"
for(int dip = 2; dip < 11; dip = dip + 1)
    print(s[dip / 2])
```

4p **Question 11.**

Write down the final value of each variable after running the following programs:

a)

```
yap = ['g', 'u', 's', 'd', 'i', 'm']
yap[4] = 's'
for(int hit = 1; hit < 2; hit = hit + 2)
    yap[hit] = yap[hit - 1]
```

b)

```
sal = ['u', 'r', 'u', 'b', 'a', 'd']
sal[5] = 'm'
for(int mux = 0; mux <= 2; mux = mux + 1)
    sal[mux] = sal[mux + 2]
```

3p **Question 12.**

Rewrite this for loop as a while loop:

```
uru = 1
for(int sal = 1; sal < 905; sal = sal + 10)
    uru += 1
```

3p **Question 13.**

Rewrite this while loop as a for loop:

```
bus = 17
while(bus > -2)
    print(bus)
    bus -= 4
```

Sort it out yourself.

Consider the following procedure (in pseudocode) for performing **bubble sort** on an array. It puts the numbers in an array in increasing order.

```

initialize counter
do
{
    reset counter to 0

    iterate through entire array
        if array[n] > array[n+1]
            swap them
            increment counter

    print array
} while(counter > 0)

```

2 p **Question 14.**

Perform the procedure on the array below, printing the array each time that we're at the bottom of the loop, like in the pseudocode above.

Remember to follow the procedure to the letter and to not take any shortcuts! Fill exactly as many rows as indicated by the print statement in the code.

[illegible][illegible]