

Introduction to Programming

Scott Morgan

@Scott3142
scott3142.com

Volunteers...

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
jump()
```

```
sleep(1)
```

```
prin('Bang!')
```

```
sleep(1)
```

```
touch_your_toes()
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
jump()
```

```
sleep(1)
```

```
prin('Bang!')
```

```
sleep(1)
```

```
touch_your_toes()
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

```
prin('Bang!')
```

Beware of spelling...



Computers are stupid... (At the moment...)

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
jump()
```

```
sleep(1)
```

```
if <next person is  
wearing glasses>:
```

```
touch_your_toes()
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

```
sleep(1)
```

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
jump()
```

```
sleep(1)
```

```
if <next person is  
wearing glasses>:
```

```
touch_your_toes()
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

```
sleep(1)
```


if <next person is wearing glasses>:

A question to the computer....

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
for <count from 1 to  
3>:
```

```
jump()
```

```
sleep(1)
```

```
touch_your_toes()
```

```
end for loop
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

```
print('Boom!')
```

```
sleep(1)
```

```
print('Crash!')
```

```
sleep(2)
```

```
print('Kapow!')
```

```
sleep(1)
```

```
for <count from 1 to  
3>:
```

```
jump()
```

```
sleep(1)
```

```
touch_your_toes()
```

```
end for loop
```

```
sleep(1)
```

```
print('Pop!')
```

```
sleep(1)
```

```
print('Beep!')
```

**for <count from 1
to 3>:**

jump()

sleep(1)

touch_your_toes()

end for loop

Repetitive actions - 'for/do' loops

Three very important concepts

Three very important concepts

Output routines: print, return

Conditionals: if, then, else if, else

Loops: for, while, do

Tick tock...

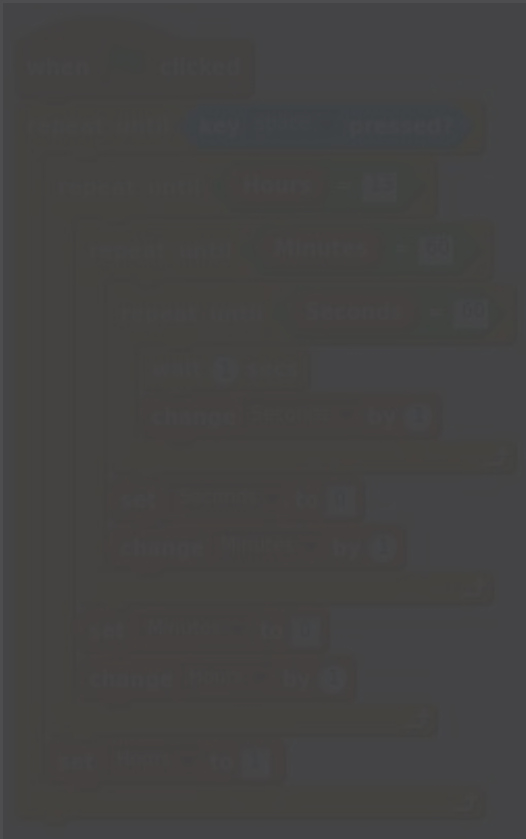


Tick tock...

Discuss the algorithm of an analogue clock.

- *How do the hands on a clock move?*
- *What happens when 60 seconds/minutes have passed?*
 - *How might you write a code to do this?*





```
when clicked
  repeat until key space pressed?
    repeat until variable 1 = 13
      repeat until x = 60
        repeat until my variable = 60
          wait 1 secs
          change my variable by 1
        set my variable to 0
        change x by 1
      set x to 0
      change variable 1 by 1
    set x to 1
```

```
when clicked
repeat until key space pressed?
repeat until Hours = 13
repeat until Minutes = 60
repeat until Seconds = 60
  wait 1 secs
  change Seconds by 1
set Seconds to 0
  change Minutes by 1
set Minutes to 0
  change Hours by 1
set Hours to 1
```

The image shows a Scratch script for a digital clock. It starts with a 'when clicked' event block. A 'repeat until' loop with the condition 'key space pressed?' contains three nested 'repeat until' loops. The innermost loop is 'repeat until Seconds = 60', which contains a 'wait 1 secs' block and a 'change Seconds by 1' block. The middle loop is 'repeat until Minutes = 60', which contains a 'set Seconds to 0' block and a 'change Minutes by 1' block. The outermost loop is 'repeat until Hours = 13', which contains a 'set Minutes to 0' block, a 'change Hours by 1' block, and a 'set Hours to 1' block.

```
when clicked
repeat until key space pressed?
repeat until variable 1 = 13
repeat until x = 60
repeat until my variable = 60
  wait 1 secs
  change my variable by 1
set my variable to 0
  change x by 1
set x to 0
  change variable 1 by 1
set x to 1
```

The image shows a Scratch script for a digital clock using custom variables. It starts with a 'when clicked' event block. A 'repeat until' loop with the condition 'key space pressed?' contains three nested 'repeat until' loops. The innermost loop is 'repeat until x = 60', which contains a 'wait 1 secs' block and a 'change my variable by 1' block. The middle loop is 'repeat until my variable = 60', which contains a 'set my variable to 0' block and a 'change x by 1' block. The outermost loop is 'repeat until variable 1 = 13', which contains a 'set x to 0' block, a 'change variable 1 by 1' block, and a 'set x to 1' block.

Three Four very important concepts

Output routines: print, return

Conditionals: if, then, else if, else

Loops: for, while, do

Variables: name appropriately!

FizzBuzz

Here is an algorithm for playing the game FizzBuzz:

- 1. Start counting from 1.*
- 2. If you reach a number in the 3 times table, say Fizz instead of the number.*
- 3. If you reach a number in the 5 times table, say Buzz instead of the number.*
- 4. If you reach a number in both the 3 and 5 times tables, say FizzBuzz.*

1. *Start counting from 1.*
2. *If you reach a number in the 3 times table, say Fizz instead of the number.*
3. *If you reach a number in the 5 times table, say Buzz instead of the number.*
4. *If you reach a number in both the 3 and 5 times tables, say FizzBuzz.*

Questions:

- *How do we check (programmatically) if a number is 'in the n times table'?*
- *How would we write this in code?*

Debugging

- *Debugging is one of the most important parts of programming.*

Debugging

- *Debugging is one of the most important parts of programming.*
- *To get good at debugging, you have to play! Make mistakes, fix them, make more mistakes.*

Debugging

- *Debugging is one of the most important parts of programming.*
- *To get good at debugging, you have to play! Make mistakes, fix them, make more mistakes.*
- *Making mistakes is how you learn! No one (EVER) gets it right the first time!*

Collatz Conjecture

Here is an algorithm for the Collatz conjecture:

- 1. Start at some natural number N .*
- 2. If the number is even, divide it by two.*
- 3. If the number is odd, multiply it by three and add one.*
- 4. Stop when you reach 1.*

NB: The conjecture is that any chain will always reach 1. It is unproven (although checked for very large positive integers).

Resources & Final Comments

Slides and Notebooks:

scott3142.com/programming

Resources & Final Comments

Slides and Notebooks:

scott3142.com/programming

*Don't worry too much about the
language you choose to learn.*

Resources & Final Comments

*Slides and Notebooks:
scott3142.com/programming*

*Don't worry too much about the
language you choose to learn.*

*There are some great tutorials and
resources around! (See webpage)*

Resources & Final Comments

*Slides and Notebooks:
scott3142.com/programming*

*Don't worry too much about the
language you choose to learn.*

*There are some great tutorials and
resources around! (See webpage)*

*~95% of programming is debugging.
Don't be afraid to make mistakes!*