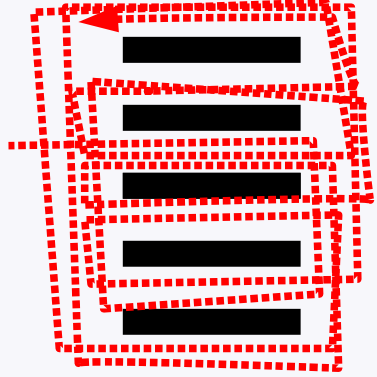
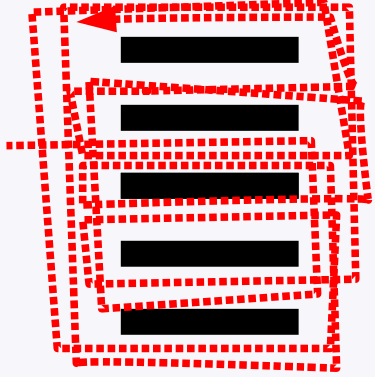


# Ways to put code into actions (instructions)

# Ways to put code into actions (instructions)



# Ways to put code into actions (instructions)

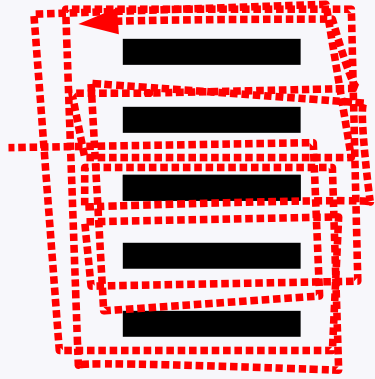


## Interpreter

Immediately  
runs

Not so efficient;  
can run slow

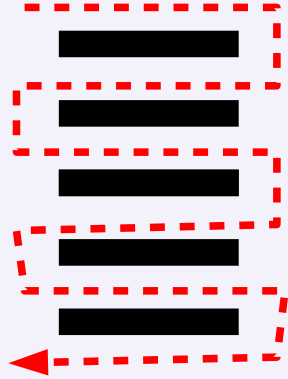
# Ways to put code into actions (instructions)



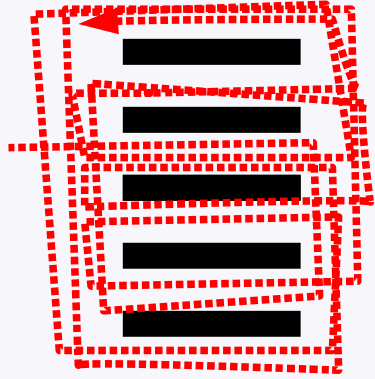
**Interpreter**

Immediately  
runs

Not so efficient;  
can run slow



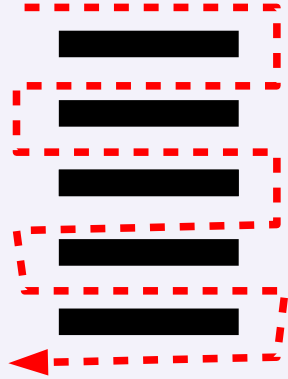
# Ways to put code into actions (instructions)



**Interpreter**

Immediately  
runs

Not so efficient;  
can run slow

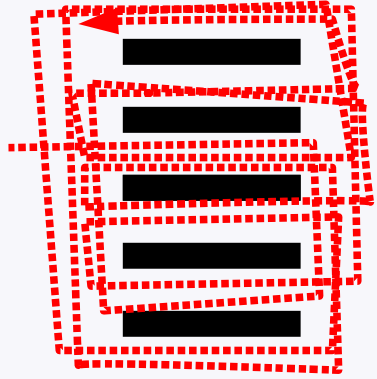


**Compiler**

Delay before  
running

Runs efficiently

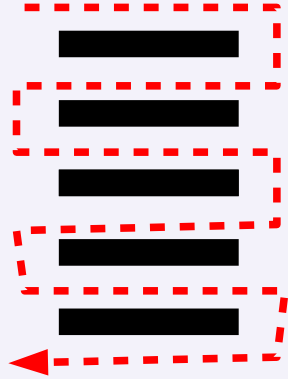
# Ways to put code into actions (instructions)



**Interpreter**

Immediately  
runs

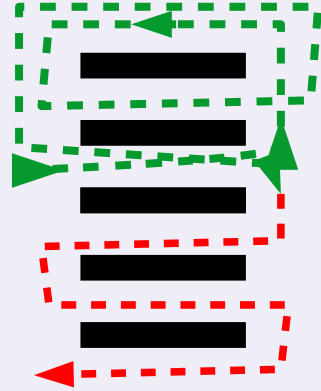
Not so efficient;  
can run slow



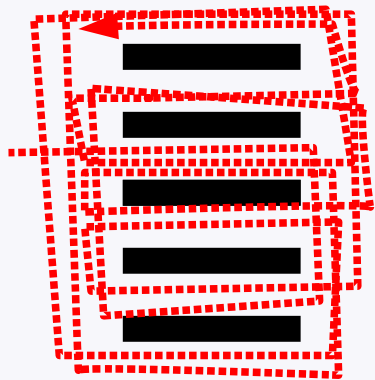
**Compiler**

Delay before  
running

Runs efficiently



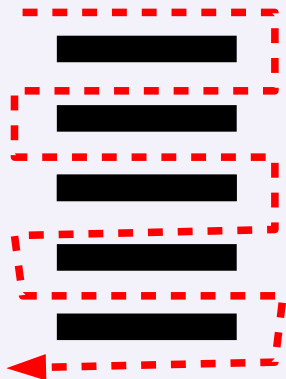
# Ways to put code into actions (instructions)



## Interpreter

Immediately  
runs

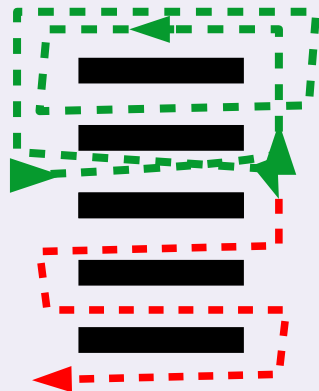
Not so efficient;  
can run slow



## Compiler

Delay before  
running

Runs efficiently

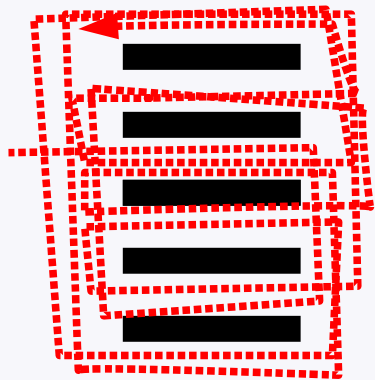


## Just-in-time

Immediately  
runs

Runs very efficiently  
( all browsers 2008)

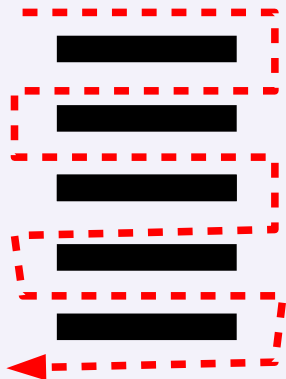
# Ways to put code into actions (instructions)



## Interpreter

Immediately  
runs

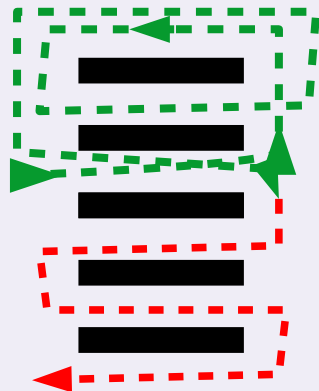
Not so efficient;  
can run slow



## Compiler

Delay before  
running

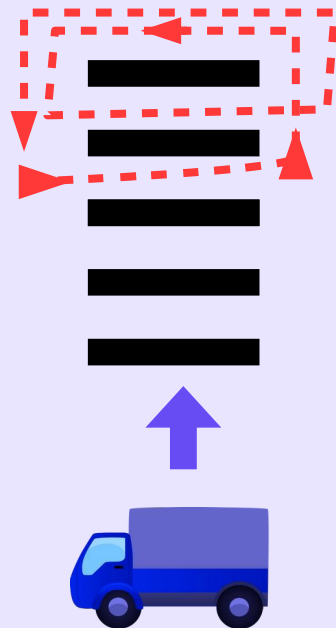
Runs efficiently



## Just-in-time

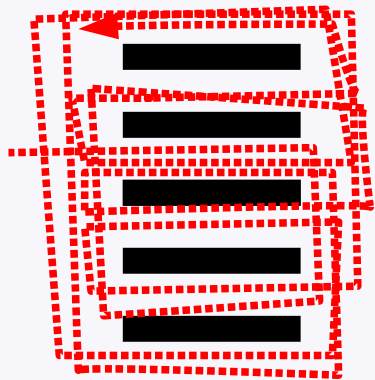
Immediately  
runs

Runs very efficiently  
( all browsers 2008)





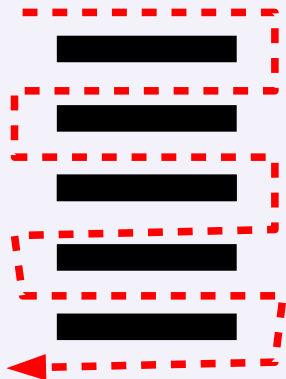
# Ways to put code into actions (instructions)



**Interpreter**

Immediately  
runs

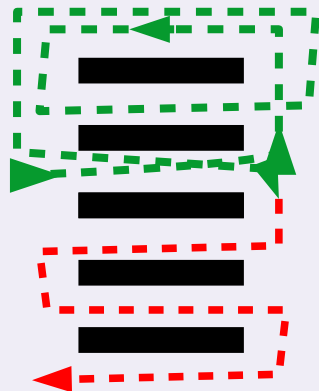
Not so efficient;  
can run slow



**Compiler**

Delay before  
running

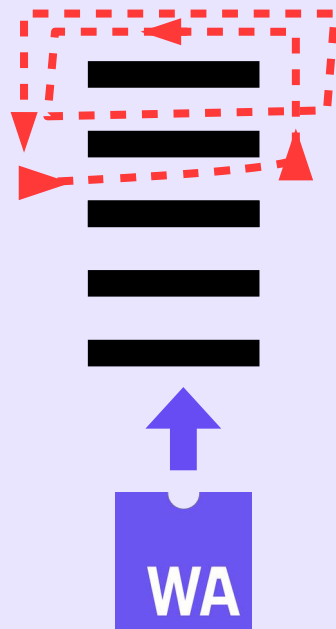
Runs efficiently



**Just-in-time**

Immediately  
runs

Runs very efficiently  
( all browsers 2008)



**WebAssembly**

Immediately  
runs highly efficiently  
(MVP 2017)

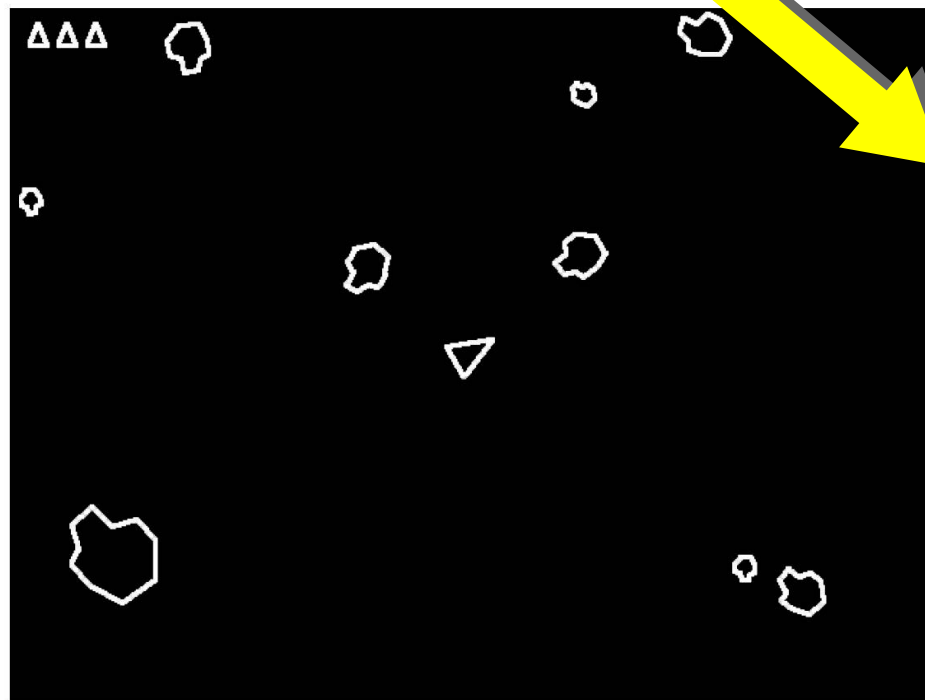
# What is WebAssembly

Asteroids in WebAssembly

[Code on GitHub](#)

An [Asteroids](#) game ported from C to WebAssembly.

(Arrows to move; Spacebar to shoot)



**Written in  
C NOT  
JavaScript**

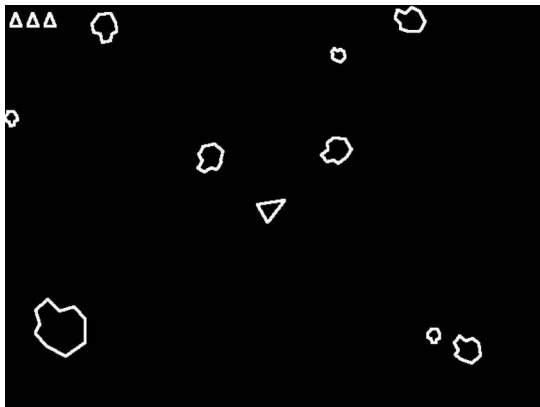
[levelupwasm.com  
/apps/asteroids](http://levelupwasm.com/apps/asteroids)

# What is WebAssembly

Asteroids in WebAssembly

Code on GitHub

An [Asteroids](#) game ported from C to WebAssembly.  
(Arrows to move, Spacebar to shoot)



Want to learn how to port games like this one to the web? Check out my book [Level Up with WebAssembly](#).

Page

Filesystem

⌵

⋮

index.html x

loading.gif

asteroids.js

▼ top

▼ www.levelupwasm.com

▼ apps/asteroids

index.html

asteroids.js

loading.gif

▶ stackpath.bootstrapcdn.com

▶ wasm

1 <!doctype html>

2 <html lang="en">

3 <head>

4 <meta charset="utf-8">

5 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

6 <meta name="description" content="Asteroids game ported from C to WebAssembly">

7 <meta name="author" content="Robert Aboukhalil">

8 <title>Asteroids Game in WebAssembly</title>

9

10 <!-- Bootstrap core CSS -->

11 <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" rel="stylesheet">

12

13 <style>

14 body { padding-top: 5rem; }

15 .starter-template { text-align: center; }

16 canvas { display: block; margin: 0 auto; }

17 </style>

18 </head>

19 <body>

20 <nav class="navbar navbar-expand-md navbar-dark bg-dark fixed-top">

21 <a class="navbar-brand" href="#">Asteroids in WebAssembly</a>

22 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">

23 <span class="navbar-toggler-icon"></span>

24 </button>

25

26 <div class="collapse navbar-collapse" id="navbarCollapse">

27 <ul class="navbar-nav mr-auto"></ul>

28 <div>

29 <a style="color: #ccc" class="nav-link" href="https://github.com/levelupwasm/levelupwasm">Level Up with WebAssembly</a>

30 </div>

31 </div>

32 </nav>

33

34 <div class="starter-template">

35 <p class="lead">

36 An <a href="https://github.com/flightcrank/asteroids">Asteroids</a> game ported from C to WebAssembly.

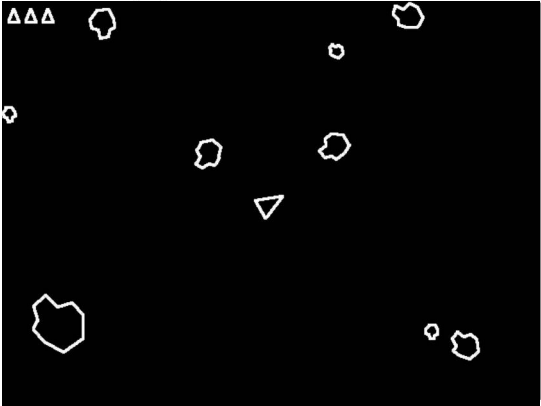
37 <small>(Arrows to move, Spacebar to shoot)</small>

# What is WebAssembly

Asteroids in WebAssembly

Code on GitHub

An [Asteroids](#) game ported from C to WebAssembly.  
(Arrows to move; Spacebar to shoot)



Want to learn how to port games like this one to the web? Check out my book [Level Up with WebAssembly](#).

Elements Console Sources Network Performance Memory Application Security Audits

Page Filesystem >> ⋮

- top
  - www.levelupwasm.com
  - stackpath.bootstrapcdn.com
  - wasm
    - wasm-0053561a
      - 0600
        - wasm-0053561a-693
        - wasm-0053561a-694
        - wasm-0053561a-695
        - wasm-0053561a-696
        - wasm-0053561a-697
        - wasm-0053561a-698
        - wasm-0053561a-699
      - 0700
      - 0800
      - 0900
      - 1000
      - 1100
      - 1200
      - 1300
      - 1400
      - 1500
      - 1600
      - 1700

wasms-0053561a-694 wasms-0053561a-696 wasms-0053561a-698 x >>

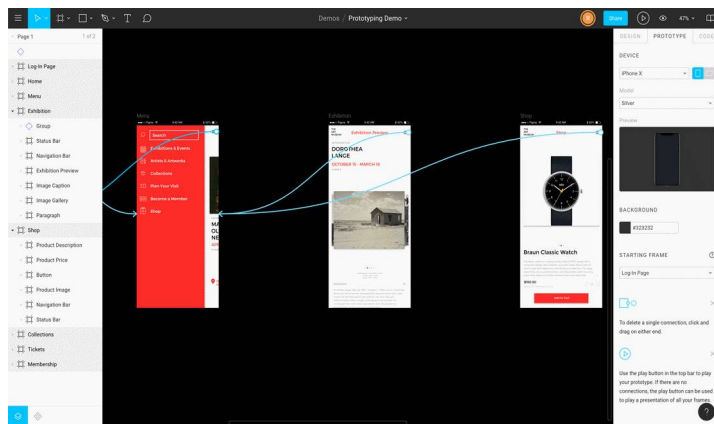
```
1 func (param i32 i32)
2 (local i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32 i32)
3   global.get 18
4   local.set 58
5   global.get 18
6   i32.const 32
7   i32.add
8   global.set 18
9   global.get 18
10  global.get 19
11  i32.get_s
12  if
13    i32.const 32
14    call 1
15  end
16  local.get 58
17  i32.const 8
18  i32.add
19  local.set 53
20  local.get 58
21  local.set 54
22  local.get 0
23  local.set 12
24  local.get 1
25  local.set 23
26  i32.const 0
27  local.set 34
28  i32.const 0
29  local.set 45
30  local.get 53
31  i32.const 23368
32  i64.load offset=0 align=4
33  i64.store offset=0 align=4
34  i32.const 0
35  local.set 34
36  loop
37    block
38      local.get 34
39      local.set 55
40      local.get 23
```

## WebAssembly code (Wasam)

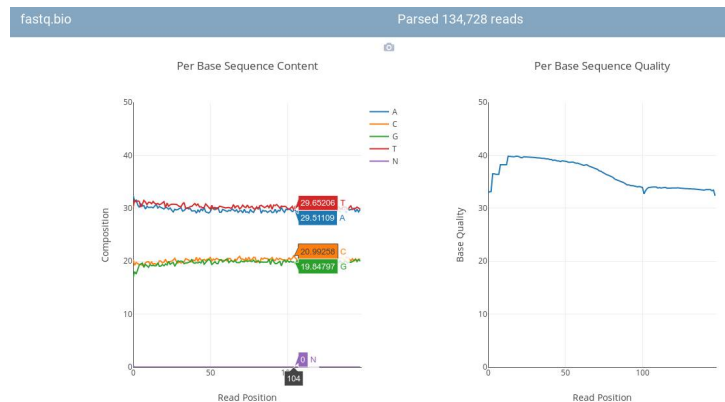
Compiled  
(converted) from C  
to WebAssembly to  
run directly in the  
browser

# Why is WebAssembly important?

*Faster than  
JavaScript*



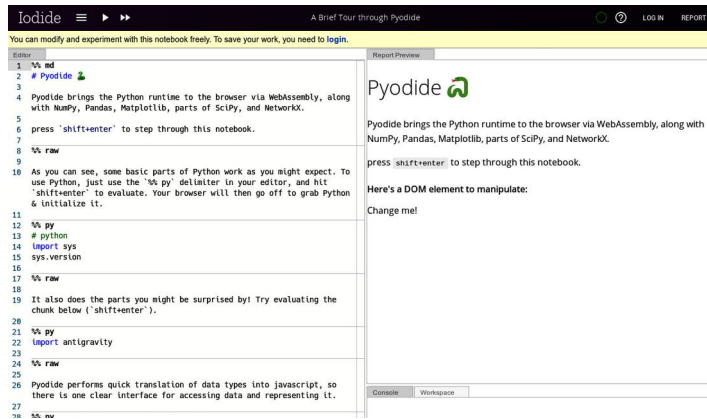
*Figma 3X faster*



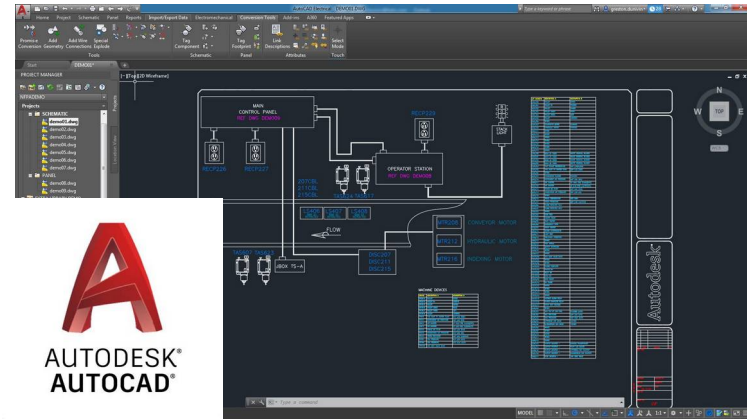
*fastq.bio DNA  
sequencer 20X faster*

# Why is WebAssembly important?

*Run established codebase*



***Pyodide is Python running on the browser via webAssembly for scientific and AI***



***AutoCAD web version uses identical codebase as desktop - C***

# TL;DR

**WebAssembly Code is 'shipped in'**  
**via a binary blob**

**Allows faster execution than**  
**JavaScript**

**Allows established non JavaScript**  
**code via compiler to run on the web**



# References

<https://hacks.mozilla.org/2017/02/a-cartoon-intro-to-webassembly/>

<https://developer.mozilla.org/en-US/docs/WebAssembly/Concepts>

<https://webassembly.org/>

<https://github.com/mbasso/awesome-wasm>

<https://github.com/robertaboukhalil/wasm-asteroids>

<https://opensource.com/article/19/4/command-line-playgrounds-webassembly>

<https://hacks.mozilla.org/2019/04/pyodide-bringing-the-scientific-python-stack-to-the-browser/>

