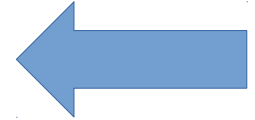# WebAssembly:

It's Cool!

What is WebAssembly?

- It's a new binary executable format for the web

- Intended to be nearly as fast as native machine code

- Not trying to replace JavaScript!

- Compile to wasm from any language you want, as long as it's C/C++ or Rust

How do you use it?

www.webassembly.org

(install the Emscripten compiler toolchain)

- Linker flag **-s WASM=1** will emit wasm

- -o myFile.js will emit the js wrapper

- Include the js wrapper file in your web project to expose the Module object

- Call your functions from JavaScript!

The Module Object

```javascript
// Call a C function from javascript:

const returnValue = Module.ccall("myCoolFunction", "number", ["number", "number"], [arg1, arg2]);
//                                                   ^ return type   ^ argument types     ^ arguments

// Interact with the C heap from javascript like this:

const pointer = Module._malloc(1000000);

// ^ allocate an array of 1000000 bytes on the heap

Module.HEAPU8.set(myImageDataObject.data, pointer);

// ^ Copy an html imagedata object to that memory location

Module._free(pointer);

// ^ Deallocate the memory!
```

It's a Sandboxed Environment

- Weird free effects: JavaScript provides free read-only C array bounds checking (I think)

- Cool for image edge handling and similar tasks

- Optional linker flags permit unrestricted memory growth – create cool memory leaks!

Demo

Does it do what it says it does?

Canvas image processing functions:

JavaScript vs. C++ showdown