Introduction to Quorum

Website: www.quorumlanguage.com

How is Quorum funded?

Quorum's initial funding started as part of a National Science Foundation project (NSF CNS-0940521). Current funding for Quorum also largely comes from the National Science Foundation (NSF CNS-1440878, NSF, DRL 1644491, and NSF DRL 1640131).

Other funders include:

- *Washington State School for the Blind*
- *Southern Illinois University Edwardsville*
- *University of Nevada, Las Vegas,*
- *Do it program at the University of Washington*
- *Reader's Digest Partners for Sight Foundation*
- *Quorum Outreach and Research Foundation*

Hour of Code Tutorial

Quorum is featured in code.org's Hour of Code in December 2014 and has had over 27,000 participants. It is the only tutorial that is fully accessible.

Quorum is open source

This means you can download the source code for the compiler or libraries or tools and modify them or contribute yourself. Quorum is under the BSD license.

It also means that **Quorum is available for all!**

About Quorum

What kind of Programming Language is Quorum?

Quorum is a general purpose tool that we call an "evidence-based" programming language. It started as a language for blind or visually impaired students. However, as Quorum gained popularity, we broadened it to work for everyone. Now, Quorum supports a wide variety of applications, like creating games, audio processing, and many other applications. It can be used in the classroom or for commercial purposes at no charge.

What are the high level technical details of Quorum?

Quorum has many features. We use evidence from experiments to make it easy to learn, but powerful and scalable. It has many features common in other languages, but has been refined for ease of use. Quorum also has a large standard library, which contains additions to the language. Examples include libraries for math, data structures, web components, physics, 2D/3D graphics, digital signal processing, 3D auditory processing, and more.

What platforms does Quorum support?

Quorum programs can be created on various versions of Windows, Mac, and in a browser. Programs written in Quorum can be compiled to run on a desktop, on the web, or on the iPhone.

What does Evidence-Based mean?

The people behind Quorum are researchers and scientists who believe in bringing evidence on how people invent software to programming language design.

Quorum started out as a project to simplify syntax and be accessible. We continue to innovate on that front, but the project has expanded to improve programming language design overall. Quorum is a fully featured and commercially scalable language.

The Quorum web site has a more detailed discussion of our evidence standards.

Learn more at: https://www.quorumlanguage.com/evidence.html

Selected Research

Stefik, A., & Siebert, S. (2013). An empirical investigation into programming language syntax. *ACM Transactions on Computing Education (TOCE), 13*(4), 19.

Stefik, A., & Hanenberg, S. (2014, October). The programming language wars: Questions and responsibilities for the programming language community. In *Proceedings of the 2014 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software* (pp. 283-299). ACM.

Quorum Overview

Libraries

Quorum 5 has a wide variety of libraries available. These libraries are designed to be easy to use and accessible.

Libraries for:

- Accessibility
- Containers
- Digital Signal Processing
- Language Features
- LEGO Robotics
- Graphics (2D and 3D)
- Mathematics
- MIDI Music
- Physics
- Sound (5.1 available)
- System Features
- Web
- Video Game Development

For Advanced Users

Quorum also supports most common programming language features like:

- Inheritance
- Generics
- Exceptions
- Plugins

Quorum is a commercial strength programming language and is written in Quorum itself. It can compile programs to desktop, the web, or to iPhone.

Basics

Primitive Types

integer – a positive or negative number or zero with no decimal point

```
integer i = 5
```

number – any real number that can also have a decimal point

```
number n = 5.3
```

boolean – a variable with two possible values: true or false

```
boolean b = false
```

text – any string of characters symbols or numbers enclosed in " "

```
text t = "Hello World!"
```

Type Conversion

Types can be converted in Quorum using the cast instructions as follows:

```
text t = "5.3"
number n = cast(number, t)
```

Variable names

Variable names must start with a letter, but can be followed by any combination of letters, numbers or underscores.

Legal:

```
number n5 = 5.3
number n_5 = 5.3
```

Illegal:

```
number 5n = 5.3
number n&$5 = 5.3
```

Operators

Math

Addition	+
Subtraction	–
Multiplication	*
Division	/
Modulus	mod

Boolean

Logical equal	=
Logical not equal	not=

Text

Concatenation	+

Statements

There are no semicolons or end of line markers needed in Quorum.

Program Control & Arrays

Conditional Statements

Simple:

```
if a = 1

end
```

If Else:

```
if a > 5

else

end
```

If Elseif:

```
if a > 10

elseif a < 10

else

end
```

Loop Statements

Repeat <expr> times:

```
repeat 10 times

end
```

Repeat while <expr>:

```
integer i = 0
repeat while i < 10

  i = i + 1
end
```

Repeat until <expr>:

```
integer i = 0
repeat until i = 10

  i = i + 1
end
```

Arrays

Arrays can be accessed by using the array library:

```
Libaries.Containers.Array
```

Fill an array:

```
Array<integer> a
integer i = 0
repeat 5 times
  a:add(i)
  i = i + 1
end
```

Print the 4th element:

```
output a:Get(3)
```

Remove the first element:

```
i = a:RemoveFromFront()
```

Sort the array:

```
a:Sort()
```

Classes and Actions

Classes

Classes are defined in a code block between the class name and an end.

```
class Dog

end
```

Actions:

Actions are like methods in other languages. The Main method must be present for a class to be runnable and execution starts there.

```
action Main
  Bark(3)
end

action Bark (integer i)
  repeat i times
    output "Bark"
  end
end
```