A general purpose programming language
with applications for science

Origins

- Created by Walter Bright (Zortech C++, Empire: Wargame of the Century etc.)
- Original compiler release in 2001 (0.00 alpha)
- Andrei Alexandrescu (C++ guru, Modern C++ Design, generic/meta-programming expert) arrives on the scene in 2004
- D2 arrives in 2007
- Last D1 release in 2012

Lineage / inspirations

- C-family syntax
- C-family semantics: If it looks like C, it works like C
- Powerful but straightforward generic/meta-programming, not like C++
- Has a (mostly) optional garbage collector
- Java-style classes (reference types, no multiple inheritance*)
- Good support for and interoperability between concepts from most leading programming paradigms

*of implementation

```c
#include <stdio.h>

int main()
{
    double x = 0.1;
    printf("%e\n", x);
    for (long i = 0; i < 20; ++i)
    {
        x = 3.7 * x * (1 - x);
        printf("%e\n", x);
    }
    return 0;
}
```

```
//#include <stdio.h>
import core.stdc.stdio;

int main()
{
    double x = 0.1;
    printf("%e\n", x);
    for (long i = 0; i < 20; ++i)
    {
        x = 3.7 * x * (1 - x);
        printf("%e\n", x);
    }
    return 0;
}
```

Logistic map calculator
========================
Prints the first n applications of the logistic map to a file.

Formula: $X\boxed{?} = r\ X\boxed{?}_{-1}\ (1 - X\boxed{?}_{-1})$

Options:
--initial=<> or -x<> : the initial value of X, i.e. X_0
--ratio=<> or -r<>   : the scaling factor r
--nSteps=<> or -n<> : the number of iterations to perform
--output=<> or -o<> : output file name

The output is in native-endian IEEE754 double-precision floats.
The output includes the initial value and therefore is 8 (nSteps + 1) bytes long.

```
print(np.arange(100000)
    .reshape((100, 1000))
    .mean(data, axis=0))
```

```
iota(100_000)
    .sliced(100, 1000)
    .transposed
    .map!mean
    .writeln;
```

Summary

- Fast by default
- Safe by default
- Flexible, low-to-zero cost abstractions
- Understandable
- Controllable

Fin