

Dlang Community Survey

The idea was to find out which parts of the language are should be improved, in which areas are working members of community and are they finding D suitable for their day-to-day work.

12 persons answered the questions.

Do you use D?

yes, frequently (multiple times per week)	7	58.3%
not anymore (I have used it previously)	3	25.0%
yes, occasionally (once every few weeks or less)	2	16.7%
no, I've never used it	0	0.0%

3 of them stop using D :(They mostly find D immature and too risky to use in production. However we all knew Symmetry and Funkwerk experience is quite good.

I currently don't use D because...

D seems too risky to use in production	2	28.6%
D seems immature; not ready for production	2	28.6%
D doesn't have the libraries I need	2	28.6%
D doesn't have enough learning materials	1	14.3%
I didn't know about D prior to this survey	0	0.0%
D doesn't solve the problem for me	0	0.0%

ОТВЕТОВ 7

How long did you use D before you stopped?

more than a year	3	100.0%
I've never used it	0	0.0%
less than a day	0	0.0%
less than a week	0	0.0%
less than a month	0	0.0%
less than a year	0	0.0%
more than 5 years	0	0.0%

ОТВЕТОВ 3

Also problems with bugs in the language and the absent of the libraries were mentioned as one of the reasons of why users abandoned D.

But there are plenty of users that like to use D.

How long have you been using D?

more than 5 years	4	44.4%
3-6 months (less than half a year)	2	22.2%
between 1 and 2 years	2	22.2%
between 2 and 5 years	1	11.1%
less than a month	0	0.0%
1-2 months	0	0.0%
7-12 months (less than a year)	0	0.0%

ОТВЕТОВ 9

What are the technical aspects or features of D you like the most?

ease of use	9	20.9%
performance (speed)	8	18.6%
meta-programming	8	18.6%
syntax	7	16.3%
standard library	6	14.0%
self-contained binaries	2	4.7%
open source code	2	4.7%
editor support	1	2.3%
FFI	0	0.0%
foreign packages	0	0.0%

ОТВЕТОВ 43

The strongest parts of D are related to ease of use, good performance and one the best meta-programming experience.

D community looks pretty welcome. Some of the users even participate in the language development.

Do you feel welcome in the D community?

yes, somewhat	5	55.6%
yes, very	2	22.2%
not at all	1	11.1%
I don't participate in the community	1	11.1%
moderate	0	0.0%

ОТВЕТОВ 9

Have you made code contributions to the D project in the past?

yes, and I plan to make more	3	33.3%
no, and I currently don't plan to	3	33.3%
no, but I plan to in the near future	2	22.2%
yes, but I won't anymore	1	11.1%

ОТВЕТОВ 9

Survey asked about some famous tools and approaches. Most of the users like Dub (but it is known that some part of the community prefers Makefiles and Meson)

dub

like	6	66.7%
dislike	1	11.1%
neither like nor dislike	1	11.1%
like very much	1	11.1%
dislike very much	0	0.0%
I have never used it	0	0.0%

betterC

like very much	3	33.3%
I have never used it	3	33.3%
dislike	2	22.2%
like	1	11.1%
dislike very much	0	0.0%
neither like nor dislike	0	0.0%

ImportC

I have never used it	5	55.6%
like	2	22.2%
neither like nor dislike	1	11.1%
like very much	1	11.1%
dislike very much	0	0.0%
dislike	0	0.0%

Most of the users prefer to write in Visual Studio Code.

What editor(s) do you use when writing D?

Visual Studio Code (VS Code)	7	63.6%
Sublime Text	2	18.2%
Emacs	1	9.1%
Other	1	9.1%
Atom	0	0.0%
IntelliJ	0	0.0%
Dexed	0	0.0%
Neovim / Vim	0	0.0%
Notepad++	0	0.0%
Visual Studio (Visual D)	0	0.0%

Areas of usage D covers all proposed cases and it is really hard to find any clear winner here.

What types of software do you develop?

automation / scripts	6	15.4%
command-line programs (CLI)	6	15.4%
data processing	5	12.8%
libraries	5	12.8%
web services	5	12.8%
games	4	10.3%
desktop programs (GUI)	2	5.1%
scientific computing	2	5.1%
embedded	2	5.1%
other	2	5.1%

The Linux obviously is the main target - there was no surprise.

What platforms are you targeting?

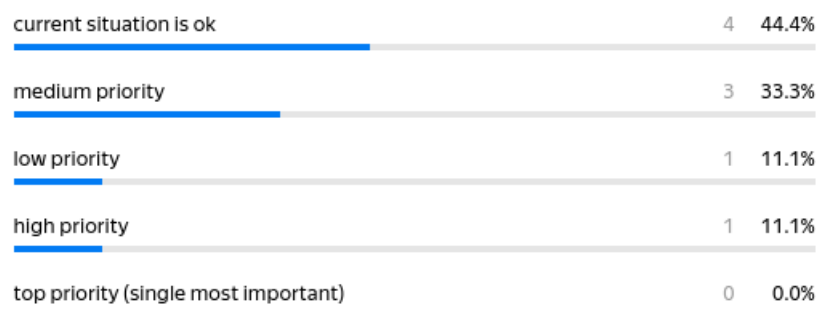
Linux	8	47.1%
Windows	5	29.4%
macOS	2	11.8%
iOS	1	5.9%
embedded	1	5.9%
BSD-variant	0	0.0%
Android	0	0.0%
JavaScript	0	0.0%
web assembly	0	0.0%
other	0	0.0%

Most of the participants of the survey regularly use: templates, threading, nogc, mixins. A little bit rarely: safe and trusted.

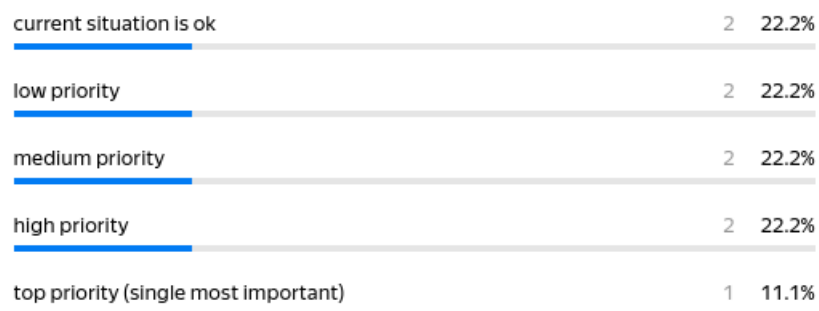
Ideas about improvement are mostly concentrated on fixing compiler bugs and improve tooling.

What should be our priorities in improving D, that would bring the largest quality-of-life improvements?

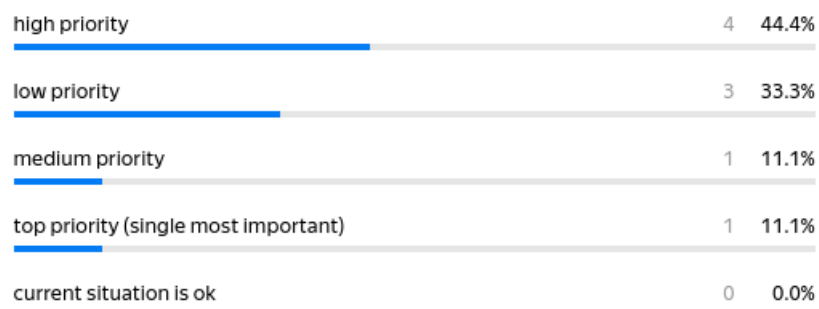
documentation

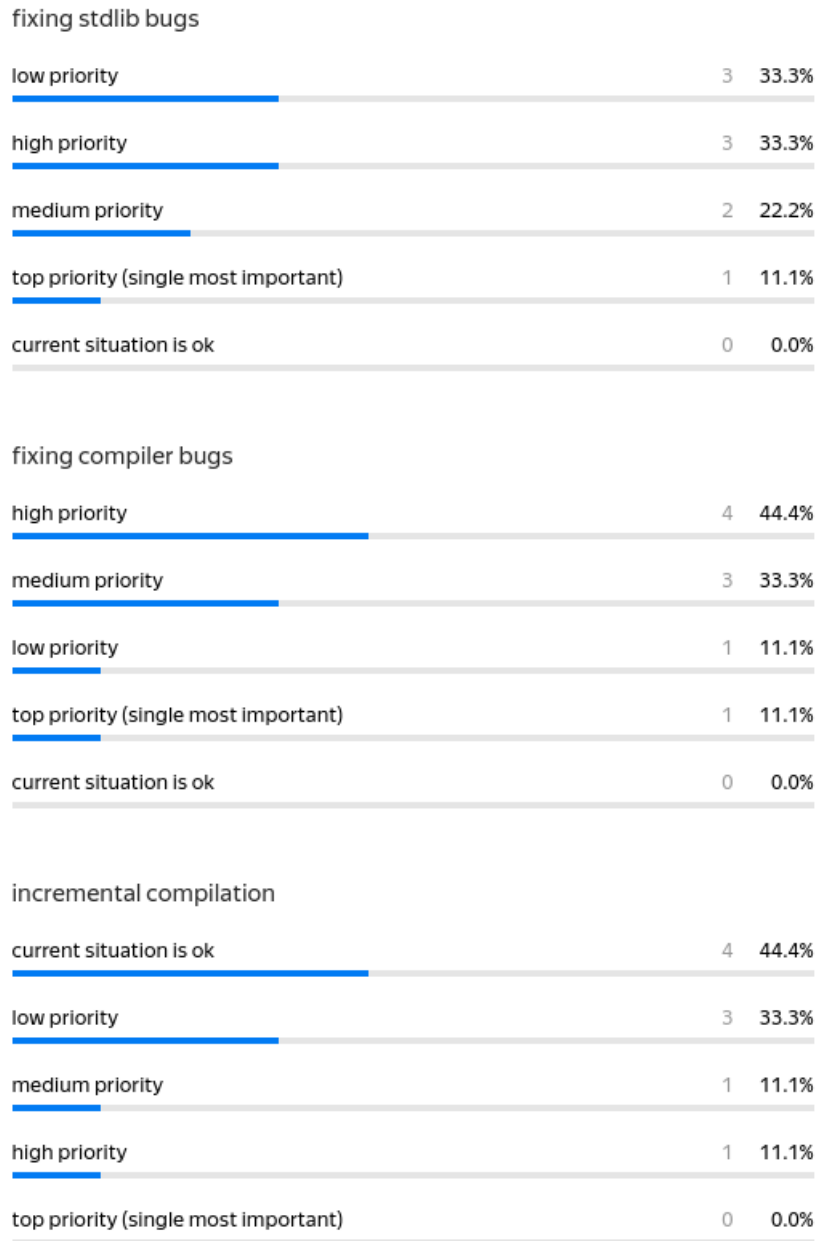


larger standard library



improve tooling





The features, tools, or libraries participants would like to have in D:

- backwards compatibility, REPL
- alias this operator overloading interfaces
- smart pointers in std, less GC in std
- improved eventloop, more robust communication libs (snmp, grpc, protobuf, etc), telemetry logging tools
- move semantics, cast operators overloading.

This survey is based on the questions from Nim lang survey. The link was shared in the one of the D telegram chat.

I want to thank all participants and all (current and previous) users of the D language.