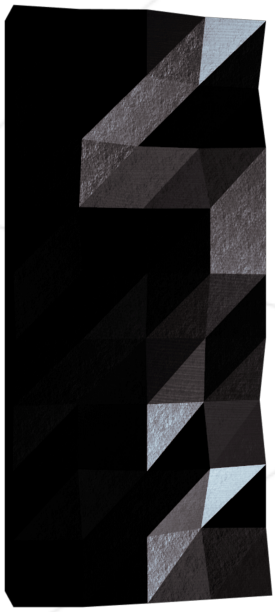

Dash - ft_yes

yes, please

*Summary: YES, we are going to make **yes***

Version: draft 2

Contents

Chapter I

Foreword

Looks simple right?

```c
#include <stdio.h>

int main()
{
        for(;;)
                printf("y\n");
}
```

Harder, Better, Faster, Stronger

Chapter II

Objective

Create the fastest ***/usr/bin/yes***.

You can ignore interactive script easily with *yes*.

```
$ yes | rm -r large_directory
```

https://en.wikipedia.org/wiki/Yes_(Unix)

Chapter III

Instructions

- Your program should never leak or unexpectedly quit(Segfault for example).
- If your program doesn't compile, it's a 0.
- We won't use norminette for dash C project.
- Evaluation will be done on 42 Seoul's Mac.
- This dash is a solo project.
- Turn in your code inside the turn-in repository.

Chapter IV

Exercice 00 : BUFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

	Exercise : 00
ft_yes	
Turn-in directory : *ex*00/	
Files to turn in : `ft_yes.c`	
Forbidden functions : `vmsplice, splice`	

- All other standard libraries and functions are allowed.
- You don't have to get arguments.
- Program should be at least faster than example code above. Otherwise it's a 0.
- Any assembly code is forbidden.

```
$ gcc -fno-asm -o ft_yes ft_yes.c
$ ./ft_yes | pv >/dev/null
136GiB 0:00:33 [4.60GiB/s] [ <=>                                        ]
```

https://en.wikipedia.org/wiki/GNU_Core_Utilities

`printf(), puts(), write()`