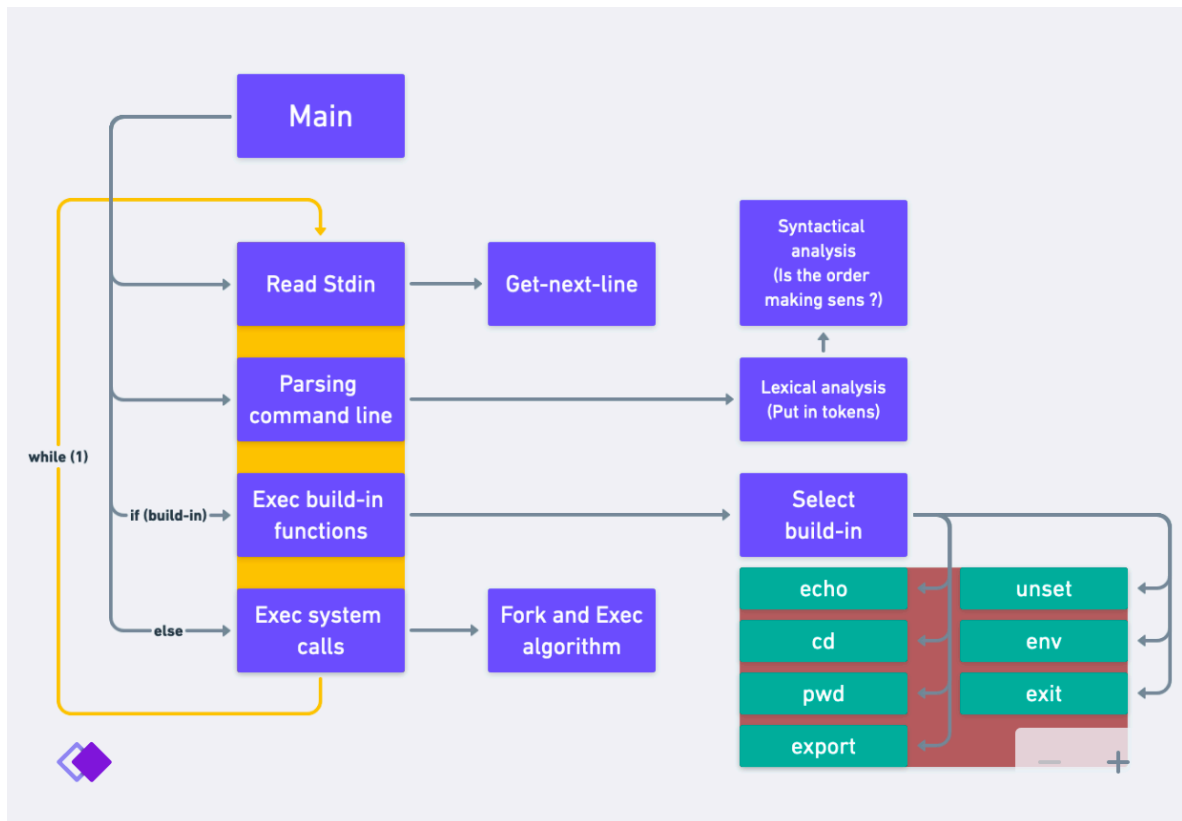


√Minishell Notes



Edge cases:

Check Pipex edge cases!

What if you unset the env PATH line?

yes

```
env | grep $PATH
```

```
echo "hello " | cat -e
```

```
ech"o" "he"llo"o"
```

```
| <command>
```

```
<command> |
```

```
<doesnotexist
```

echo 123 > out | echo 456 >> out —> first left should be executed, then the right part (try couple of times)

```
cat << e1 > x << e2 > y << e3 >> x
```

```
cat << e1 > x << e2 > y <"<" e3 >> x
echo >>>test
```

For more details, please visit <https://support.apple.com/kb/HT208050>.

```
bash-3.2$ echo >>>test
bash: syntax error near unexpected token `>'
bash-3.2$ echo >>>>test
bash: syntax error near unexpected token `>>'
bash-3.2$ echo >>>>>test
bash: syntax error near unexpected token `>>'
bash-3.2$ echo <<<test
bash-3.2$ echo <<<<test
bash: syntax error near unexpected token `<'
bash-3.2$ echo <<<<<test
bash: syntax error near unexpected token `<<'
bash-3.2$ echo <<<<<<test
bash: syntax error near unexpected token `<<<'
bash-3.2$ echo <<<<<<<test
bash: syntax error near unexpected token `<<<'
bash-3.2$ █
```

```
$> e"cho hello"
00110001: echo hello: command not found
```

Execution:

Receive an array containing the command(s).

Use getenv to get the path, parse this like in Pipex and run through all possibilities.

Create a pipe and a fork, use execve on the child processes.

3:28

> always the last one will be considered as the input file:
<in1 grep h <in2

In this example in1 will be ignored and in2 will be read from.

This:

```
<"i"n"1" grep h
```

Is equal to this:

```
<in1 grep h
```

Open returns -1

Pipe = void

```
>asd4 >asd5 cat >asd6 out >asd1 >asd2 >asd3
```

Search for >, everything after that is a filename

I used an enum structure and defined the macros there

5:08

its in my mylib

5:08

under libft.h

5:08

5:09

```
# ifndef T_FILELST
#  define T_FILELST
enum e_redirection_mode
{
    REDIR_VOID,
    REDIR_IN,
    REDIR_OUT,
    REDIR_APPEND,
    REDIR_HEREDOC,
    REDIR_NONE
};
typedef struct s_filelst
{
    char                *filename;
    enum e_redirection_mode    mode;
```

```
    struct s_filelst    *next;
}    t_filelst;
# endif
```

(edited)

5:10

The whole thing looks like that. I didnt use the filelst structure I think, only the enums

5:11

I used REDIR_NONE if there is no real redirection

5:11

REDIR_IN for '<'

REDIR_OUT for '>'

5:12

REDIR_HEREDOC for '<<'

REDIR_APPEND for '>>'

REDIR_VOID if the input for the simple command is just an immediate EOF

5:12

For example:

```
cat -e Makefile <djaiisadjdas | wc -l
```

5:12

the file does not exist for the first command so it wont output anything. The second simple command wc -l will get REDIR_VOID as its input

5:13

so an EOF

New

5:17

I used a trick to send EOF

5:17

```
ret = dup(STDIN_FILENO);
    dup2(pipeTmp[0], STDIN_FILENO);
    close(pipeTmp[0]);
    dup2(ret, STDIN_FILENO);
    close(ret);
    dup2(ret, STDIN_FILENO);
    close(ret);
```

5:17

it looks like that

5:19

in my mind it kind of makes sense. What it does is replace STDIN with an fd and then replace it back, kind of like a temporary file

**thats created that has nothing inside it, so it acts like an EOF.
Didnt find other ways to do this behavior. And ofc since you are
replacing STDIN with another fd, you'll have to restore it,
which ret does in this case**

CD:

| | |
|------------------------------|-----------------|
| cd | --> working |
| cd .. | --> working |
| cd . | --> working |
| cd /Users | --> working |
| cd // | --> working |
| cd '/' | --> working |
| cd ///// | --> working |
| cd ./././ | --> working |
| cd / | --> working |
| cd '/////' 2>/dev/null | --> NOT working |
| cd '/etc' | --> working |
| cd '/var' | --> working |
| cd "\$PWD/file_tests" | --> NOT working |
| cd "doesntexist" | --> working |
| cd "doesntexist" 2>/dev/null | --> working |
| cd ../../.. | --> working |
| cd .. | --> working |
| cd .. | --> working |
| cd ? | --> working |
| cd + | --> working |
| cd _ | --> working |
| cd bark bark | --> working |
| cd '/' | --> working |
| cd \$PWD/file_tests | --> NOT working |
| cd \$OLDPWD/something | --> NOT working |
| cd ~ | --> NOT working |

ECHO:

| | |
|-----------|-------------|
| echo | --> working |
| echo echo | --> working |
| eCho | --> working |

```

eChO                                --> working
eCHO                                --> working
ECHO                                --> working
echo rhobebou                       --> working
echo stop barking                   --> working
echo "bonjour"                      --> working
echo bonjour                        --> working
echo 'bonjour'                     --> working
echo -n bonjour                    --> working
echo -nn bonjour                   --> working
echo -n -n -n bonjour              --> working
echo "-n" bonjour                  --> working
echo -n"-n" bonjour                --> NOT working
echo "-nnnn" bonjour              --> working
echo "-n -n -n"-n bonjour         --> NOT working
echo "-n '-n'" bonjour            --> NOT working
echo ~                             --> NOT working
echo "~"                           --> NOT working
echo '~'                           --> NOT working
echo ~123                          --> NOT working
echo 123~                          --> NOT working
echo ~/123                        --> NOT working
echo ~/123/456                    --> NOT working
echo $USER                        --> working
echo "$USER"                      --> NOT working
echo "'$USER'"                   --> NOT working
echo " '$USER' "                  --> NOT working
echo text"$USER"                  --> NOT working
echo text"$USER" ' $USER '
echo "text" "$USER" "$USER"
echo '      $USER      '
echo      text "$USER"      "$USER"text
echo ""$USER""
echo ""$USER""
echo $USER'$USER'text oui oui    oui oui $USER oui    $USER "
echo $USER " $USER $USER " $USER " $USER -n $USER
echo '\ ' \
echo "\" ' \" \" \"
echo \\\" \\\" \\\" \\\"\\\"\\\" \\\"\\\"\\\"
echo "$USER""$USER""$USER"
echo text"$USER"test
echo '$USER' "$USER" "text \ ' text"
echo '$USER'
echo $USER " "
echo "$USER""Users/$USER/file""$USER""$USER'
echo "$USER$USER$USER"

```

```
echo '$USER'$USER'$USER'
echo '$USER'$USER'$USER'$USER'
echo " $USER '$PWD'
echo " $USER \$ '$PWD'
echo $USER=4
echo $USER=thallard
echo $USER
echo $?
echo $USER213
echo $USER$12USER$USER=4$USER12
echo $USER $123456789USER $USER123456789
echo $USER $9999USER $8888USER $7777USER
echo $USER $USER9999 $USER8888 $USER7777
echo $USER $USER9999 $USER8888 $USER7777 "$USER"
echo "$USER=12$USER"
echo "$9USER" "$USER=12$SOMETHING"
echo $PWD/file
echo "$PWD/file"
echo "text" "text$USER" ... "$USER"
echo $PWD
```

EXIT:

```
exit 0 0
exit 42 42
exit -42 -24
exit 42
exit 42 53 68
exit 259
exit -12030
exit --1239312
exit ++++1203020103
exit +0
exit ++++++0
exit -----0
exit azerty
exit kewkwqke
exit a
exit z
exit "1"
exit "2"
exit "+102"
exit "1230"
exit "+++1230"
exit "1"23
exit "2"32"32"
```

```
exit ""42""
exit '42'42"42
exit +'42'42"42
exit -'42'42"42
exit 9223372036854775807
exit 9223372036854775808
exit -4
exit wrong
exit wrong_command
exit something
exit 1
exit -1
exit 42
exit 0
exit --000
exit +++++++000
exit ++++3193912939
exit ---31232103012
exit "something"
exit q
exit help
exit export
exit echo
exit cd ..
exit 0 0
exit something somethingv2
exit 42 42 42 42 42
exit echo something
exit exit
```

EXPORT:

```
env | grep "_="
export | grep "SHLVL"
export | grep "OLDPWD"
export | grep "PWD"
export $?
export TEST
export TEST=
export TEST=123
export __TEST=123
export --TEST=123
export ""=""
export ""=""
export ""=""=""
export '='=''
```



```
export TE\\ST=100
export TE-ST=100
export -TEST=100
export TEST-=100
export _TEST=100
export | grep "SHLVL"
export TEST
export =====
export 1TEST=
export TEST
export ""=""
export TES=T=""
export TE+S=T=""
export TES\\T=123
export TES.T=123
export TES\\$T=123
export TES\\T
export TES.T=123
export TES+T=123
export TES=T=123
export TES}T=123
export TES{T=123
export TES-T=123
export -TEST=123
export _TEST=123
export TES_T=123
export TEST_=123
export TE*ST=123
export TES#T=123
export TES@T=123
export TES!T=123
export TES$?T=123
export =====123
export ++++++=123
export _____=123
export export
export echo
export pwd
export cd
export export
export unset
export sudo
export TES^T=123
export TES!T=123
export TES\\~T=123
export TEST+=100
```

PIPES:

```
env | grep "_="
env | grep "SHLVL"
echo oui | cat -e
echo oui | echo non | echo something | grep oui
echo oui | echo non | echo something | grep non
echo oui | echo non | echo something | grep something
cd .. | echo "something"
cd .. | echo "something"
cd / | echo "something"
cd .. | pwd
ifconfig | grep ":"
ifconfig | grep nothing
whoami | grep $USER
whoami | grep $USER > tmp/file
whoami | cat -e | cat -e > tmp/file
cat Makefile | grep "FLAGS"
cat Makefile | cat -e | cat -e
cat Makefile | grep "FLAGS" | grep "FLAGS" | cat -e
export TEST=123 | cat -e | cat -e
unset TEST | cat -e
echo test | cat -e | cat -e | cat -e
whereis ls | cat -e | cat -e > test
echo test | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat
-e | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat -e | cat
-e | cat -e
ls -la | grep "."
whereis grep > tmp/file
whereis grep > tmp/file
ls -la > tmp/file
ls -la > tmp/file
```

UNSET:

```
unset
export TEST=100
unset doesntexist
unset PWD
unset PWD
unset OLDPWD
unset PATH
unset PATH
unset PATH
unset TES\\T
```

```
unset TES;T
unset TES.T
unset TES+T
unset TES=T
unset TES}T
unset TES{T
unset TES-T
unset -TEST
unset _TEST
unset TES_T
unset TEST_
unset TE*ST
unset TES#T
unset TES@T
unset TES!T
unset TES$?T
unset =====
unset +++++++
unset _____
unset export
unset echo
unset pwd
unset cd
unset unset
unset sudo
unset TES^T
unset TES!T
unset TES\~T
```