MINISHELL

Gestion des comportements

test.c		A faire	A tester	Leaks ?	Fonctionne	Comportement attendu	Sortie d'exit (echo $?)	Notes
		X	X	OK KO	OK KO			
$> \n (touche entrée)				OK	OK	$>	Exit de l'ancienne commande	
$> [que des espaces]				OK	OK		0	
$> [que des tabulations]				OK	OK		0	Crl + V + Tab
$> :				OK	OK		0	Marche aussi pour "#"
$> !				OK	OK		1	
$> >				OK	OK	bash: syntax error near unexpected token `newline' $>	2	
$> <				OK	OK		2	
$> >>				OK	OK		2	
$> <<				OK	OK		2	
$> <>				OK	OK		2	
$> >>>>>				OK	OK	bash: syntax error near unexpected token `>>' $>	2	
$> >>>>>>>>>>>>>>				OK	OK	bash: syntax error near unexpected token `>>' $>	2	
$> <<<<<				OK	OK	bash: syntax error near unexpected token `<<' $>	2	
$> <<<<<<<<<<<<<<				OK	OK	bash: syntax error near unexpected token `<<' $>	2	
$> > > > >				OK	OK	bash: syntax error near unexpected token `>' $>	2	Marche aussi pour <
$> >> >> >> >>				OK	OK	bash: syntax error near unexpected token `>>' $>	2	Marche aussi pour <
$> >>>> >> >> >>				OK	OK	bash: syntax error near unexpected token `>>' $>	2	Marche aussi pour <
$> /				OK	OK	bash: /: Is a directory $>	126	
$> //				OK	OK	bash: //: Is a directory $>	126	
$> /.				OK	OK	bash: /.: Is a directory $>	126	
$> /./../../../../..				OK	OK	bash: /./../../../../..: Is a directory $>	126	
$> ////////				OK	OK	bash: ////////: Is a directory $>	126	
$> \\		NON GERE		OK	OK	bash: \: command not found	127	
$> \\\\		NON GERE		OK	OK	bash: \\: command not found	127	
$> \\\\\\\\		NON GERE		OK	OK	bash: \\\\: command not found	127	
$> -				OK	OK	bash: -: command not found $>	127	Marche aussi pour "_@"]=+£$?"
$> |				OK	OK	bash: syntax error near unexpected token `|' $>	2	Marche aussi pour les metacharacters "&();"
$> | hola				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> | | |				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> ||				OK	OK	bash: syntax error near unexpected token `||' $>	2	
$> |||||				OK	OK	bash: syntax error near unexpected token `||' $>	2	
$> ||||||||||||				OK	OK	bash: syntax error near unexpected token `||' $>	2	
$> >>|><				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> &&				OK	OK	bash: syntax error near unexpected token `&&' $>	2	
$> &&&&&				OK	OK	bash: syntax error near unexpected token `&&' $>	2	
$> &&&&&&&&&&&&&				OK	OK	bash: syntax error near unexpected token `&&' $>	2	
$> ;;		NON GERE		OK	OK	bash: syntax error near unexpected token `;;' $>	2	
$> ;;;;;;		NON GERE		OK	OK	bash: syntax error near unexpected token `;;' $>	2	
$> ;;;;;;;;;;;;;;;;		NON GERE		OK	OK	bash: syntax error near unexpected token `;;' $>	2	
$> ()		NON GERE		OK	OK	bash: syntax error near unexpected token `)' $>	2	
$> (( ))		NON GERE		OK	OK	bash: syntax error near unexpected token `)' $>	2	
$> (((( ))))		NON GERE		OK	OK	bash: syntax error near unexpected token `)' $>	2	
$> ""		NON GERE		OK	OK	bash: : command not found $>	127	
$> "hola"				OK	OK	bash: hola: command not found $>	127	
$> 'hola'				OK	OK	bash: hola: command not found $>	127	
$> ''		NON GERE		OK	OK	bash: : command not found $>	127	
$> *		NON GERE		OK	OK	bash: crashtest.c: command not found $>	127	[BONUS]
$> */*		NON GERE		OK	OK	bash: Docs/attest.cation: command not found $>	127	[BONUS]

CARACTERES A LA VOLEE (SYNTAXE)

Command					Output	Code	Notes
$> */*	NON GERE			OK	OK	bash: */*: No such file or directory $>	127	**[BONUS]** S'il n'y aucun dossier à l'intérieur
$> .	NON GERE			OK	OK	bash: .: filename argument required $>	2	
$> ..	NON GERE			OK	OK	bash: ..: command not found $>	127	
$> ~	NON GERE			OK	OK	bash: /home/vietdu91: Is a directory $>	126	
$> ABC=hola	NON GERE			OK	OK	$>	0	/!\ A gerer comme une erreur
$> 4ABC=hola	NON GERE			OK	OK	bash: 4ABC=hola: command not found $>	127	
hola				OK	OK	bash: hola: command not found $>	127	
$> hola que tal				OK	OK	bash: hola: command not found $>	127	
$> Makefile				OK	OK	bash: Makefile: command not found $>	127	
$> echo				OK	OK	$>	0	
$> echo -n				OK	OK	$>	0	
$> echo Hola				OK	OK	Hola $>	0	
$> echoHola				OK	OK	bash: echoHola: command not found $>	127	
$> echo-nHola				OK	OK	bash: echo-nHola: command not found $>	127	
$> echo -n Hola				OK	OK	Hola$>	0	
$> echo "-n" Hola				OK	OK	Hola$>	0	
$> echo -nHola				OK	OK	-nHola $>	0	
$> echo Hola -n				OK	OK	Hola -n $>	0	
$> echo Hola Que Tal				OK	OK	Hola Que Tal $>	0	
$> echo Hola				OK	OK	Hola $>	0	
$> echo Hola Que Tal				OK	OK	Hola Que Tal $>	0	
$> echo \n hola				OK	OK	n hola $>	0	A gerer ?
$> echo " " | cat -e				OK	OK	$ $>	0	
$> echo | cat -e				OK	OK	$ $>	0	
$> """'echo hola"""''' que"""' tal""''				OK	OK	hola que tal $>	0	
$> echo -n -n				OK	OK	$>	0	
$> echo -n -n Hola Que				OK	OK	Hola Que$>	0	
$> echo -p				OK	OK	-p $>	0	
$> echo -nnnnn				OK	OK	$>	0	
$> echo -n -nnnn -nnnn				OK	OK	$>	0	
$> echo -n-nnn -nnnn				OK	OK	-n-nnn -nnnn $>	0	
$> echo -n -nnn hola -nnnn				OK	OK	hola -nnnn$>	0	
$> echo -n -nnn-nnnn				OK	OK	-nnn-nnnn$>	0	
$> echo --------n				OK	OK	--------n $>	0	
$> echo -nnn --------n				OK	OK	--------n$>	0	
$> echo -nnn -----nn---nnnn				OK	OK	-----nn---nnnn$>	0	
$> echo -nnn --------nnnn				OK	OK	--------nnnn$>	0	
$> echo $				OK	OK	$ $>	0	
$> echo $?				OK	OK	0 $>	0	Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
$> echo $?$				OK	OK	0$ $>	0	
$> echo $? | echo $? | echo $?				OK	OK	0 $>	0	
$> echo $:$= | cat -e				OK	OK	$:$=$ $>	0	
$> echo " $ " | cat -e				OK	OK	$ $ $>	0	
$> echo ' $ ' | cat -e				OK	OK	$ $ $>	0	
$> echo $HOME				OK	OK	/home/vietdu91 $>	0	
$> echo \$HOME				OK	OK	$HOME $>	0	
$> echo my shit terminal is [$TERM]				OK	OK	my shit terminal is [xterm-256color] $>	0	
$> echo my shit terminal is [$TERM4				OK	OK	my shit terminal is [$>	0	
$> echo my shit terminal is [$TERM4]				OK	OK	my shit terminal is [] $>	0	
$> echo $UID				OK	OK	1000 $>	0	
$> echo $HOME9				OK	OK	$>	0	Marche pour tous les caractères alphanumériques de l'ASCII
$> echo $9HOME				OK	OK	HOME $>	0	
$> echo $HOME%				OK	OK	/home/vietdu91% $>	0	Marche pour les autres caractères (ponctuation, caractères non-ASCII, etc.)
$> echo UIDHOME				OK	OK	1000/home/vietdu91% $>	0	
$> echo Le path de mon HOME est $HOME				OK	OK	Le path de mon HOME est /home/vietdu91 $>	0	
$> echo $USER$var\$USER$USER\$USERtest$USER				OK	OK	vietdu91$USERvietdu91$USERtestvietdu91 $>	0	A ne pas gerer si '\' non gere : Doit sortir : vietdu91vietdu91vietdu91vietdu91
$> echo $hola*				OK	OK	$>	0	Cas d'un variable inexistant
$> echo -nnnn $hola				OK	OK	$>	0	
$> echo > <				OK	OK	bash: syntax error near unexpected token `<' $>	2	
$> echo | |				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> EechoE				OK	OK	bash: EechoE: command not found $>	127	
$> .echo.				OK	OK	bash: .echo.: command not found $>	127	
$> >echo>				OK	OK	bash: syntax error near unexpected token `newline' $>	2	
$> <echo<				OK	OK	bash: syntax error near unexpected token `newline' $>	2	
$> >>echo>>				OK	OK	bash: syntax error near unexpected token `newline' $>	2	
$> |echo|				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> |echo -n hola				OK	OK	bash: syntax error near unexpected token `|' $>	2	
$> echo *	NON GERE			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c $>	0	**[BONUS]**
$> echo '*'	NON GERE			OK	OK	* $>	0	Marche aussi pour les double quotes
$> echo D*	NON GERE			OK	OK	Docs Dockers Drawings $>	0	**[BONUS]**
$> echo *Z	NON GERE			OK	OK	*Z $>	0	**[BONUS]**
$> echo *t hola	NON GERE			OK	OK	crashtest.c test.c hola $>	0	**[BONUS]**
$> echo *t	NON GERE			OK	OK	crashtest.c test.c $>	0	**[BONUS]**
$> echo $*	NON GERE			OK	OK	$>	0	**[BONUS]**
$> echo hola*hola *	NON GERE			OK	OK	hola*hola crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c $>	0	**[BONUS]**
$> echo $hola*	NON GERE			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c $>	0	**[BONUS]**
$> echo $HOME*	NON GERE			OK	OK	/home/vietdu91 $>	0	**[BONUS]**
$> echo $""				OK	OK	$>	0	
$> echo "$"""				OK	OK	$ $>	0	
$> echo '$'''				OK	OK	$ $>	0	

Commande				OK	OK	Sortie	Code	Commentaire
`$> echo $"HOME"`				OK	OK	HOME $>	0	
`$> echo $''HOME`				OK	OK	HOME $>	0	
`$> echo $""HOME`				OK	OK	HOME $>	0	
`$> echo "$HO"ME`				OK	OK	ME $>	0	
`$> echo '$HO'ME`				OK	OK	$HOME $>	0	
`$> echo "$HO""ME"`				OK	OK	ME $>	0	
`$> echo '$HO''ME'`				OK	OK	$HOME $>	0	
`$> echo "'$HO''ME"`				OK	OK	''ME' $>	0	Il y a bien 3 singles quotes avant le ME. Je ne sais pas pourquoi Excel tronque à l'affichage le 3e quote
`$> echo ""$HOME`				OK	OK	/home/vietdu91 $>	0	
`$> echo "" $HOME`				OK	OK	/home/vietdu91 $>		
`$> echo ''$HOME`				OK	OK	/home/vietdu91 $>	0	
`$> echo '' $HOME`				OK	OK	/home/vietdu91 $>		
`$> echo $"HO""ME"`				OK	OK	HOME $>	0	
`$> echo $'HO''ME'`				OK	OK	HOME $>	0	
`$> echo $'HOME'`				OK	OK	HOME $>	0	
`$> echo "$"HOME`				OK	OK	$HOME $>	0	
`$> echo $=HOME`				OK	OK	$=HOME $>	0	
`$> echo $"HOLA"`				OK	OK	HOLA $>	0	
`$> echo $'HOLA'`				OK	OK	HOLA $>	0	
`$> echo $DONTEXIST Hola`				OK	OK	Hola $>	0	Le cas ou la variable n'existe pas et n'est pas importee
`$> echo "hola"`				OK	OK	hola $>	0	
`$> echo 'hola'`				OK	OK	hola $>	0	
`$> echo ''hola''`				OK	OK	hola $>	0	
`$> echo ''h'o'la''`				OK	OK	hola $>	0	
`$> echo "''h'o'la''"`				OK	OK	'h'o'la'' $>	0	
`$> echo ""h'o'la""`				OK	OK	hola' $>	0	
`$> echo"'hola'"`				OK	OK	bash: echo'hola': command not found $>	0	
`$> echo "'hola'"`				OK	OK	'hola' $>	0	
`$> echo '"hola"'`				OK	OK	"hola" $>	0	
`$> echo '''ho''''l"a'''`				OK	OK	ho"l"a $>	0	
`$> echo hola"""""""""""`				OK	OK	hola $>	0	
`$> echo hola"'''''''''''`				OK	OK	hola''''''''''' $>	0	
`$> echo hola'''''''''''`				OK	OK	hola $>	0	
`$> echo hola'"""""""""""'`				OK	OK	hola""""""""""" $>	0	
`$> e"cho hola"`				OK	OK	bash: echo hola: command not found $>	127	
`$> e'cho hola'`				OK	OK	bash: echo hola: command not found $>	127	
`$> echo "hola     " | cat -e`				OK	OK	hola $ $>	0	
`$> echo ""hola`				OK	OK	hola $>	0	
`$> echo "" hola`				OK	OK	hola $>	0	
`$> echo ""          hola`				OK	OK	hola $>	0	
`$> echo ""hola`				OK	OK	hola $>	0	
`$> echo "" hola`				OK	OK	hola $>	0	
`$> echo hola""bonjour`				OK	OK	holabonjour $>	0	
`$> "e"'c'ho 'b'"o"nj"o"'u'r`				OK	OK	bonjour $>	0	
`$> ""e"'c'ho 'b'"o"nj"o"'u'r"`				OK	OK	bash: e'c'ho 'b'onjo'u'r: command not found $>	127	
`$> echo "$DONTEXIST"Makefile`				OK	OK	Makefile $>	0	
`$> echo "$DONTEXIST""Makefile"`				OK	OK	Makefile $>	0	
`$> echo "$DONTEXIST" "Makefile"`				OK	OK	Makefile $>	0	Attention a l'espace avant
`$> $?`				OK	OK	bash: 0: command not found $>	127	Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
`$> $?$?`				OK	OK	bash: 00: command not found $>	127	
`$> ?$HOME`				OK	OK	bash: ?/home/vietdu91: command not found $>	127	
`$> $`				OK	OK	bash: $: command not found $>	127	
`$> $HOME`				OK	OK	bash: /home/vietdu91: Is a directory $>	126	
`$> $HOMEdskjhfkdshfsd`				OK	OK	$>	0	
`$> "$HOMEdskjhfkdshfsd"`				OK	OK	bash: : command not found $>	127	
`$> '$HOMEdskjhfkdshfsd'`				OK	OK	bash: $HOMEsddfddfssfdfs: command not found $>	127	
`$> $DONTEXIST`				OK	OK	$>	0	Cas d'un variable inexistant
`$> $LESS$VAR`				OK	OK	bash: -R: command not found $>	127	
`$> Ctlr-C`				OK	OK	$> ^C	130	
`$> holaCtlr-C`				OK	OK	$> hola^C	130	
`$> cat (faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])`				OK	OK	 ^C $>	130	
`$> cat | rev(faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])`				OK	OK	 ^C $>	0	
`$> Ctlr-D`				OK	OK	$> exit ~/minishell $>	0	Sort du programme
`$> hola Ctlr-D`				OK	OK	$> hola	/	Ne fait rien
`$> Ctlr-\`				OK	OK	$>	/	Ne fait rien
`$> hola Ctlr-\`				OK	OK	$> hola	/	
`$> cat (faire Ctlr-\ apres avoir fait plusieurs fois [ENTREE])`				OK	OK	 ^\Quit (core dumped) $>	131	
`$> Ctlr-Z`				OK	OK	$>	/	Ne fait rien
`$> sleep 3 | sleep 3 | sleep 3 (faire Ctlr-C une seconde apres)`				OK	OK	^C $>	130	
`$> sleep 3 | sleep 3 | sleep 3 (faire Ctlr-D une seconde apres)`				OK	OK	$>	0	Ne fait rien
`$> sleep 3 | sleep 3 | sleep 3 (faire Ctlr-\ une seconde apres)`				OK	OK	^\Quit (core dumped) $>	131	
`$> env`				OK	OK	... PATH=... $>	0	
`$> env hola`	NON GERE			OK	OK	env: 'hola': No such file or directory $>	127	
`$> env hola que tal`	NON GERE			OK	OK	env: 'hola': No such file or directory $>	127	

Commande						Sortie	Code		Commentaire

`$> env env`	NON GERE			OK	OK	`...`
						`PATH=...`
						`$>`
							0		Affiche une seule fois env

`$> env env env env env`	NON GERE			OK	OK	`...`
						`PATH=...`
						`$>`
							0		Affiche une seule fois env

`$> env ls`	NON GERE			OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
						`philosophers srcs test.c`
						`$>`
							0		

`$> env ./Makefile`	NON GERE			OK	OK	`env: './Makefile': Permission denied` `$>`	126		

`$> export HOLA=bonjour`
`$> env`
				OK	OK	`...`
						`HOLA=bonjour`
						`$>`
							0
							0		

`$> export       HOLA=bonjour`
`$> env`
				OK	OK	`...`
						`HOLA=bonjour`
						`$>`
							0
							0		

`$> export`				OK	OK	`...`
						`export PATH="..."`
						`$>`
							0		Les variables sont triees par ordre alphabetique On n'oublie pas les doubles quotes qui designent les strings attribues a chaque variable

`$> export Hola`
`$> export`
				OK	OK	`...`
						`export HOME="/home/vietdu91"`
						`export Hola`
						`export PATH="..."`
						`$>`
							0
							0
									Hola n'existe pas dans env mais il existe dans export

`$> export Hola9hey`
`$> export`
				OK	OK	`...`
						`export HOME="/home/vietdu91"`
						`export Hola9hey`
						`export PATH="..."`
						`$>`
							0
							0		

`$> export $DONTEXIST`				OK	OK	`...`
						`export HOME="/home/vietdu91"`
						`export PATH="..."`
						`$>`
							0		Le cas ou la variable n'existe pas et n'est pas importee

`$> export	grep "HOME"`				OK	OK	`export HOME="/home/vietdu91"` `$>`	0		
`$> export ""`				OK	OK	`bash: export: '': not a valid identifier` `$>`	1		
`$> export =`				OK	OK	`bash: export: '=': not a valid identifier` `$>`	1		
`$> export %`				OK	OK	`bash: export: '%': not a valid identifier` `$>`	1		
`$> export $?`				OK	OK	`bash: export: '0': not a valid identifier` `$>`	1		
`$> export ?=2`				OK	OK	`bash: export: '?=2': not a valid identifier` `$>`	1		
`$> export 9HOLA=`				OK	OK	`bash: export: '9HOLA=': not a valid identifier` `$>`	1		

`$> export HOLA9=bonjour`
`$> env`
				OK	OK	`...`
						`HOLA9=bonjour`
						`$>`
							0
							0		

`$> export _HOLA=bonjour`
`$> env`
				OK	OK	`...`
						`_HOLA=bonjour`
						`$>`
							0
							0		

`$> export ___HOLA=bonjour`
`$> env`
				OK	OK	`...`
						`___HOLA=bonjour`
						`$>`
							0
							0		

`$> export _HO_LA_=bonjour`
`$> env`
				OK	OK	`...`
						`_HO_LA_=bonjour`
						`$>`
							0
							0		

`$> export HOL@=bonjour`				OK	OK	`bash: export: 'HOL@=bonjour': not a valid identifier` `$>`	1		Cas où le nom de variable ne contient pas de lettres, ni de chiffres, ni d'underscores
`$> export HOL\~A=bonjour`	NON GERE			OK	OK	`bash: export: 'HOL~A=bonjour': not a valid identifier` `$>`	1		
`$> export -HOLA=bonjour`				OK	OK	`bash: export: -H: invalid option`	2		
`$> export --HOLA=bonjour`				OK	OK	`bash: export: --: invalid option`	2		
`$> export HOLA-=bonjour`				OK	OK	`bash: export: 'HOLA-=bonjour': not a valid identifier` `$>`	1		
`$> export HO-LA=bonjour`				OK	OK	`bash: export: 'HO-LA=bonjour': not a valid identifier` `$>`	1		
`$> export HOL.A=bonjour`				OK	OK	`bash: export: 'HOL.A=bonjour': not a valid identifier` `$>`	1		
`$> export HOL\\\$A=bonjour`	NON GERE			OK	OK	`bash: export: 'HOL$A=bonjour': not a valid identifier` `$>`	1		
`$> export HO\\\\LA=bonjour`	NON GERE			OK	OK	`bash: export: 'HO\\LA=bonjour': not a valid identifier` `$>`	1		
`$> export HOL}A=bonjour`				OK	OK	`bash: export: 'HOL}A=bonjour': not a valid identifier` `$>`	1		
`$> export HOL{A=bonjour`				OK	OK	`bash: export: 'HOL{A=bonjour': not a valid identifier` `$>`	1		
`$> export HO*LA=bonjour`				OK	OK	`bash: export: 'HO*LA=bonjour': not a valid identifier` `$>`	1		
`$> export HO#LA=bonjour`				OK	OK	`bash: export: 'HO#LA=bonjour': not a valid identifier` `$>`	1		
`$> export HO@LA=bonjour`				OK	OK	`bash: export: 'HO@LA=bonjour': not a valid identifier` `$>`	1		
`$> export HO!LA=bonjour`				OK	OK	`bash: !LA=bonjour: event not found`	0		

`$> export HO$?LA=bonjour`
`$> env`
				OK	OK	`...`
						`HO0LA=bonjour`
						`$>`
							0		

`$> export +HOLA=bonjour`				OK	OK	`bash: export: '+HOLA=bonjour': not a valid identifier` `$>`	1		
`$> export HOL+A=bonjour`				OK	OK	`bash: export: 'HOL+A=bonjour': not a valid identifier` `$>`	1		

`$> export HOLA+=bonjour`
`$> env`
	NON GERE			OK	OK	`...`
						`HOLA=bonjour`
						`$>`
							0		

`$> export HOLA=bonjour`
`$> export HOLA+=bonjour`
`$> env`
	NON GERE			OK	OK	`...`
						`HOLA=bonjourbonjour`
						`$>`
							0		"+=" s'ajoute au contenu de la variable

`$> exportHOLA=bonjour`
`$> env`
	NON GERE			OK	OK	`...`
						`PATH=...`
						`$>`
							0		

`$> export HOLA =bonjour`				OK	OK	`bash: export: '=bonjour': not a valid identifier` `$>`	1		
`$> export HOLA = bonjour`				OK	OK	`bash: export: '=': not a valid identifier` `$>`	1		

`$> export HOLA=bon jour`
`$> env`
				OK	OK	`...`
						`HOLA=bon`
						`$>`
							0		

`$> export HOLA= bonjour`
`$> env`
				OK	OK	`...`
						`HOLA=`
						`$>`
							0		

`$> export HOLA=bonsoir`
`$> export HOLA=bonretour`
`$> export HOLA=bonjour`
`$> env`
				OK	OK	`...`
						`HOLA=bonjour`
						`$>`
							0
							0
							0		

`$> export HOLA=$HOME`
`$> env`
				OK	OK	`...`
						`HOLA=/home/vietdu91`
						`$>`
							0
							0		

`$> export HOLA=bonjour$HOME`
`$> env`
				OK	OK	`...`
						`HOLA=bonjour/home/vietdu91`
						`$>`
							0
							0		

`$> export HOLA=$HOMEbonjour`
`$> env`
				OK	OK	`...`
						`HOLA=`
						`$>`
							0
							0		

`$> export HOLA=bon$jour`
`$> env`
				OK	OK	`...`
						`HOLA=bon`
						`$>`
							0
							0
									Le cas où la variable "$jour" n'existe pas

`$> export HOLA=bon\jour`
`$> env`
	NON GERE			OK	OK	`...`
						`HOLA=bonjour`
						`$>`
							0
							0		

`$> export HOLA=bon\\jour`
`$> env`
	NON GERE			OK	OK	`...`
						`HOLA=bon\jour`
						`$>`
							0
							0		

`$> export HOLA=bon(jour`	NON GERE			OK	OK	`bash: syntax error near unexpected token '('` `$>`	2		Marche pour les caractères : ()
`$> export HOLA=bon()jour`	NON GERE			OK	OK	`bash: syntax error near unexpected token '('` `$>`	2		
`$> export HOLA=bon&jour`	NON GERE			OK	OK	`bash: jour: command not found` `$>`	127		Marche pour les caractères : &|

`$> export HOLA=bon@jour`
`$> env`
				OK	OK	`$> env`
						`...`
						`HOLA=bon@jour`
						`$>`
							0		

`$> export HOLA=bon;jour`
`$> env`
				OK	OK	`bash: jour: command not found`
						`$> env`
						`...`
						`HOLA=bon`
						`$>`
							127
							0		

`$> export HOLA=bon!jour`				OK	OK	`bash: !jour: event not found` `$>`	1		

`$> export HOLA=bon"jour"`
`$> env`
				OK	OK	`...`
						`HOLA=bonjour`
						`PATH=...`
						`$>`
							0
							0		

Commande					Sortie	Code	Commentaire

`$> export HOLA$USER=bonjour`
`$> env`
			OK	OK	`...`
					`HOLAvietdu91=bonjour`
					`PATH=...`
					`$>`
						0
						0	

`$> export HOLA=bonjour=casse-toi` `$> echo $HOLA`			OK	OK	`HOLA=bonjour=casse-toi` `$>`	0	
`$> export "HOLA=bonjour"=casse-toi` `$> echo $HOLA`			OK	OK	`HOLA=bonjour=casse-toi` `$>`	0	

`$> export HOLA=bonjour`
`$> export BYE=casse-toi`
`$> echo $HOLA et $BYE`
			OK	OK	`bonjour et casse-toi`
					`$>`
						0
						0
						0	

`$> export HOLA=bonjour BYE=casse-toi` `$> echo $HOLA et $BYE`			OK	OK	`bonjour et casse-toi` `$>`	0 0	

`$> export $HOLA=bonjour`
`$> env`
			OK	OK	`...`
					`bonjour=bonjour`
					`PATH=...`
					`$>`
						0	

`$> export HOLA="bonjour        "` `$> echo $HOLA	cat -e`			OK	OK	`bonjour$` `$>`	0	
`$> export HOLA="   -n bonjour    "` `$> echo $HOLA`			OK	OK	`bonjour$>`	0	
`$> export HOLA="bonjour    "/` `$> echo $HOLA`			OK	OK	`bonjour /` `$>`	0	
`$> export HOLA=''` `$> echo " $HOLA "	cat -e`			OK	OK	`"  $` `$>`	0	
`$> export HOLA=at` `$> c$HOLA Makefile`			OK	OK	`[cela affiche le Makefile]`	0	

`$> export "" HOLA=bonjour`
`$> env`
			OK	OK	`bash: export: `"`: not a valid identifier`
					`$> env`
					`...`
					`HOLA=bonjour`
					`$>`
						1	

`$> export HOLA="cat Makefile	grep NAME"` `$> echo $HOLA`			OK	OK	`cat Makefile	grep NAME` `$>`	0	
`$> export HOLA=hey` `$> echo $HOLA$HOLA$HOLA=hey$HOLA`			OK	OK	`heyheyhey=heyhey` `$>`	0	
`$> export HOLA="  bonjour   hey   "` `$> echo $HOLA	cat -e`			OK	OK	`bonjour hey$` `$>`	0 0	
`$> export HOLA="  bonjour   hey   "` `$> echo """$HOLA"""	cat -e`			OK	OK	`   bonjour    hey    $` `$>`	0 0	
`$> export HOLA="  bonjour   hey   "` `$> echo wesh"$HOLA"	cat -e`			OK	OK	`wesh   bonjour    hey    $` `$>`	0 0	
`$> export HOLA="  bonjour   hey   "` `$> echo wesh"$HOLA".`			OK	OK	`wesh bonjour hey .` `$>`	0 0	/!\ Bien supprimer les espaces en trop au debut et a la fin du string d'export
`$> export HOLA="  bonjour   hey   "` `$> echo wesh$""HOLA.`			OK	OK	`weshHOLA.` `$>`	0 0	
`$> export HOLA="  bonjour   hey   "` `$> echo wesh$"HOLA HOLA".`			OK	OK	`weshHOLA HOLA.` `$>`	0 0	

`$> export HOLA=bonjour`
`$> export HOLA=" hola et $HOLA"`
`$> echo $HOLA`
			OK	OK	`hola et bonjour`
					`$>`
						0
						0
						0	

`$> export HOLA=bonjour`
`$> export HOLA=' hola et $HOLA'`
`$> echo $HOLA`
			OK	OK	`hola et $HOLA`
					`$>`
						0
						0
						0	

`$> export HOLA=bonjour`
`$> export HOLA=" hola et $HOLA"$HOLA`
`$> echo $HOLA`
			OK	OK	`hola et bonjourbonjour`
					`$>`
						0
						0
						0	

`$> export HOLA="ls        -l    - a"` `$> echo $HOLA`			OK	OK	`ls -l - a` `$>`	0 0	

`$> export HOLA="s -la"`
`$> l$HOLA`
			OK	OK	`total 16`
					`drwx------  2 vietdu91 [...] .`
					`drwxrwxewt 51 root     [...] ..`
					`-rwxr-xr-x  8 vietdu91 [...] crashtest.c`
					`[...]`
					`$>`
						0
						0	

`$> export HOLA="s -la"` `$> l"$HOLA"`			OK	OK	`bash: ls -la: command not found` `$>`	127	
`$> export HOLA="s -la"` `$> l'$HOLA'`			OK	OK	`bash: l$HOLA: command not found` `$>`	127	
`$> export HOLA="l"` `$> $HOLAs`			OK	OK	`$>`	0	

`$> export HOLA="l"`
`$> "$HOLA"s`
			OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
					`philosophers  srcs  test.c`
					`$>`
						0	

`$> export HOL=A=bonjour`
`$> env`
			OK	OK	`...`
					`HOL=A=bonjour`
					`$>`
						0	

`$> export HOLA="l"` `$> '$HOLA's`			OK	OK	`bash: $HOLAs: command not found` `$>`	127	

`$> export HOL=A=""`
`$> env`
			OK	OK	`...`
					`HOL=A=`
					`$>`
						0
						0	

`$> export TE+S=T` `$> env`			OK	OK	`bash: export: `TE+S=T`: not a valid identifier` `$>`	1	
`$> export ""=""`			OK	OK	`bash: export: `=`: not a valid identifier` `$>`	1	
`$> export ''=''`			OK	OK	`bash: export: `=`: not a valid identifier` `$>`	1	
`$> export "="="="`			OK	OK	`bash: export: `===`: not a valid identifier` `$>`	1	
`$> export '='='='`			OK	OK	`bash: export: `===`: not a valid identifier` `$>`	1	

`$> export HOLA=p`
`$> export BYE=w`
`$> $HOLA"BYE"d`
			OK	OK	`bash: pBYEd: command not found`
					`$>`
						127	

`$> export HOLA=p`
`$> export BYE=w`
`$> "$HOLA"'$BYE'd`
			OK	OK	`bash: p$BYEd: command not found`
					`$>`
						127	

`$> export HOLA=p`
`$> export BYE=w`
`$> "$HOLA""$BYE"d`
			OK	OK	`/home/vietdu91/42_works/minishell`	0	

`$> export HOLA=p`
`$> export BYE=w`
`$> $"HOLA"$"BYE"d`
			OK	OK	`bash: HOLABYEd: command not found`
					`$>`
						127	

`$> export HOLA=p`
`$> export BYE=w`
`$> $'HOLA'$'BYE'd`
			OK	OK	`bash: HOLABYEd: command not found`
					`$>`
						127	

`$> export HOLA=-n` `$> "echo $HOLA" hey`			OK	OK	`bash: echo -n: command not found` `$>`	127	
`$> export A=1 B=2 C=3 D=4 E=5 F=6 G=7 H=8` `$> echo "$A'$B'"$C"$D'$E'"$F'"'$G'$H"`			OK	OK	`1'2$C"$D5"$F'7'8`	0 0	

`$> export HOLA=bonjour`
`$> env`
`$> unset HOLA`
`$> env`
			OK	OK	`...`
					`PATH=...`
					`$>`
						0
						0
						0
						0	

`$> export HOLA=bonjour`
`$> env`
`$> unset HOLA`
`$> unset HOLA`
`$> env`
			OK	OK	`...`
					`PATH=...`
					`$>`
						0
						0
						0
						0
						0	

`$> unset PATH` `$> echo $PATH`			OK	OK	`$>`	0 0	
`$> unset PATH` `$> ls`			OK	OK	`bash: ls: No such file or directory` `$>`	127	
`$> unset ""`			OK	OK	`bash: unset: `': not a valid identifier` `$>`	1	
`$> unset INEXISTANT`			OK	OK	`$>`	0	Cas où la variable n'existe pas
`$> unset PWD` `$> env	grep PWD` `$> pwd`			OK	OK	`$>` `$>` `/home/vietdu91/42_works/minishell` `$>`	0 0 0	
`$> pwd` `$> unset PWD` `$> env	grep PWD` `$> cd $PWD` `$> pwd`			OK	OK	`/home/vietdu91/42_works/minishell` `$>` `$>` `$>` `/home/vietdu91`	0 0 0 0 0	
`$> unset OLDPWD` `$> env	grep OLDPWD`			OK	OK	`$>` `$>`	0 0	
`$> unset 9HOLA`			OK	OK	`bash: unset: `9HOLA`: not a valid identifier` `$>`	1	
`$> unset HOLA9`			OK	OK	`$>`	0	
`$> unset HOL?A`			OK	OK	`bash: unset: `HOL?A`: not a valid identifier` `$>`	1	
`$> unset HOLA HOL?A`			OK	OK	`bash: unset: `HOL?A`: not a valid identifier` `$>`	1	Il unset bien HOLA
`$> unset HOL?A HOLA`			OK	OK	`bash: unset: `HOL?A`: not a valid identifier` `$>`	1	Il unset bien HOLA

`$> unset HOL?A HOL.A`			OK	OK	`bash: unset: `HOL?A`: not a valid identifier`
					`bash: unset: `HOL.A`: not a valid identifier`
					`$>`
						1	

`$> unset HOLA=`			OK	OK	`bash: unset: `HOLA=`: not a valid identifier` `$>`	1	
`$> unset HOL\\\\A`	NON GERE		OK	OK	`bash: unset: `HOL\\A`: not a valid identifier` `$>`	1	
`$> unset HOL;A`	NON GERE		OK	OK	`bash: A: command not found`	127	
`$> unset HOL.A`			OK	OK	`bash: unset: `HOL.A`: not a valid identifier`	1	
`$> unset HOL+A`			OK	OK	`bash: unset: `HOL+A`: not a valid identifier`	1	
`$> unset HOL=A`			OK	OK	`bash: unset: `HOL=A`: not a valid identifier`	1	
`$> unset HOL{A`			OK	OK	`bash: unset: `HOL{A`: not a valid identifier`	1	

	Commande		OK	OK	Sortie	Code	Commentaire
FICHIERS BINAIRES[1]	`$> unset HOL}A`		OK	OK	bash: unset: `HOL}A': not a valid identifier	1	
	`$> unset HOL-A`		OK	OK	bash: unset: `HOL-A': not a valid identifier	1	
	`$> unset -HOLA`		OK	OK	bash: unset: -H: invalid option	2	
	`$> unset _HOLA`		OK	OK	$>	0	
	`$> unset HOL_A`		OK	OK	$>	0	
	`$> unset HOLA_`		OK	OK	$>	0	
	`$> unset HOL*A`		OK	OK	bash: unset: `HOL*A': not a valid identifier	1	
	`$> unset HOL#A`		OK	OK	bash: unset: `HOL#A': not a valid identifier	1	
	`$> unset $HOLA`		OK	OK	$>	0	Il va unset le contenu de $HOLA, c'est-a-dire la variable bonjour
	`$> unset $PWD`		OK	OK	bash: unset: `/home/vietdu91/42_works/minishell': not a valid identifier	1	
	`$> unset HOL@`		OK	OK	bash: unset: `HOL@': not a valid identifier	1	
	`$> unset HOL!A`		OK	OK	bash: !A: event not found	1	
	`$> unset HOL^A`		OK	OK	bash: unset: `HOL^A': not a valid identifier	1	
	`$> unset HOL$?A`		OK	OK	$>	0	Unset prend en compte le contenu de $?, c'est-a-dire qu'il lit HOL0A ou HOL127A
	`$> unset HOL\~A`		OK	OK	bash: unset: `HOL~': not a valid identifier	1	
	`$> unset "" HOLA` `$> env | grep HOLA`		OK	OK	bash: unset: `': not a valid identifier $> $>	1 0	HOLA sera quand meme unset !!
	`$> unset PATH` `$> echo $PATH`		OK	OK	$>	0 0	
	`$> unset PATH` `$> cat Makefile`		OK	OK	bash: cat: No such file or directory $>	0 1	
	`$> unset =`		OK	OK	bash: unset: `=': not a valid identifier	0	
	`$> unset ======`		OK	OK	bash: unset: `======': not a valid identifier	0	
	`$> unset ++++++`		OK	OK	bash: unset: `++++++': not a valid identifier	0	
	`$> unset _____`		OK	OK	$>	0	
	`$> unset export`		OK	OK	$>	0	
	`$> unset echo`		OK	OK	$>	0	
	`$> unset pwd`		OK	OK	$>	0	
	`$> unset cd`		OK	OK	$>	0	
	`$> unset unset`		OK	OK	$>	0	
	`$> unset sudo`		OK	OK	$>	0	
	`$> export hola | unset hola | echo $?`		OK	OK	0 $>	0	
	`$> /bin/echo`		OK	OK	$>	0	
	`$> /bin/echo Hola Que Tal`		OK	OK	Hola Que Tal $>	0	
	`$> /bin/env`		OK	OK	... PATH=... $>	0	
	`$> /bin/cd Desktop`		OK	OK	bash: /bin/cd: No such file or directory	127	
HISTORIQUE	`$> history`	NON GERE	OK	OK	... 558 echo $? 559 clear 560 history $>	0	Affiche les 500 dernières commandes
	`$> [touche du haut]`		OK	OK	$> clear	/	Defile l'historique
PWD	`$> pwd`		OK	OK	/home/vietdu91/42_works/minishell $>	0	Retourne $PWD
	`$> pwd hola`		OK	OK	/home/vietdu91/42_works/minishell $>	0	
	`$> pwd ./hola`		OK	OK	/home/vietdu91/42_works/minishell $>	0	
	`$> pwd hola que tal`		OK	OK	/home/vietdu91/42_works/minishell $>	0	
	`$> pwd -p`		OK	OK	bash: pwd: -p: invalid option $>	2	
	`$> pwd --p`		OK	OK	bash: pwd: --: invalid option $>	2	
	`$> pwd ---p`		OK	OK	bash: pwd: --: invalid option $>	2	
	`$> pwd -- p`	NON GERE	OK	OK	/home/vietdu91/42_works/minishell $>	0	
	`$> pwd pwd pwd`		OK	OK	/home/vietdu91/42_works/minishell $>	0	N'affiche qu'une fois
	`$> pwd ls`		OK	OK	/home/vietdu91/42_works/minishell $>	0	
	`$> pwd ls env`		OK	OK	/home/vietdu91/42_works/minishell $>	0	
CD	`$> cd`		OK	OK	$>	0	On doit se retrouver dans le dossier défini par $HOME /home/vietdu91
	`$> cd .`		OK	OK	$>	0	L'user reste dans le même dossier /home/vietdu91/42_works/minishell
	`$> cd ./`		OK	OK	$>	0	/home/vietdu91/42_works/minishell
	`$> cd ./././.`		OK	OK	$>	0	/home/vietdu91/42_works/minishell
	`$> cd ./././/`		OK	OK	$>	0	/home/vietdu91/42_works/minishell
	`$> cd ..`		OK	OK	$>	0	On se retrouve au dossier précédent /home/vietdu91/42_works
	`$> cd ../`		OK	OK	$>	0	/home/vietdu91/42_works
	`$> cd ../..`		OK	OK	$>	0	/home/vietdu91
	`$> cd ../.`		OK	OK	$>	0	/home/vietdu91/42_works
	`$> cd ../././.`		OK	OK	$>	0	/home/vietdu91/42_works
	`$> cd srcs`		OK	OK	$>	0	/home/vietdu91/42_works/minishell/srcs
	`$> cd srcs objs`		OK	OK	bash: cd: too many arguments $>	1	/home/vietdu91/42_works/minishell
	`$> cd 'srcs'`		OK	OK	$>	0	/home/vietdu91/42_works/minishell/srcs
	`$> cd "srcs"`		OK	OK	$>	0	/home/vietdu91/42_works/minishell/srcs
	`$> cd '/etc'`		OK	OK	$>	0	Marche aussi avec les double quotes /
	`$> cd /e'tc'`		OK	OK	$>	0	/etc
	`$> cd /e"tc"`		OK	OK	$>	0	/etc
	`$> cd sr`		OK	OK	bash: cd: sr: No such file or directory $>	1	
	`$> cd Makefile`		OK	OK	bash: cd: Makefile: Not a directory $>	1	
	`$> cd ../minishell`		OK	OK	$>	0	/home/vietdu91/42_works/minishell
	`$> cd ../../../../../..`		OK	OK	$>	0	/
	`$> cd ../../../../../../bin/ls`		OK	OK	bash: cd: ../../../../../../bin/ls: No such file or directory $>	1	/home/vietdu91/42_works/minishell
	`$> cd /`		OK	OK	$>	0	/
	`$> cd '/'`		OK	OK	$>	0	Marche aussi avec les double quotes /
	`$> cd //` `$> pwd`	NON GERE	OK	OK	// $>	0	/
	`$> cd '//'` `$> pwd`	NON GERE	OK	OK	// $>	0	Marche aussi avec les double quotes /
	`$> cd ///` `$> pwd`		OK	OK	/ $>	0	/
	`$> cd ////////` `$> pwd`		OK	OK	/ $>	0	/
	`$> cd '////////'` `$> pwd`		OK	OK	/ $>	0	Marche aussi avec les double quotes /
	`$> cd /minishell`		OK	OK	bash: cd: /minishell: No such file or directory $>	1	
	`$> cd /` `$> cd ..`		OK	OK	$>	0	/
	`$> cd _`		OK	OK	bash: cd: _: No such file or directory $>	1	
	`$> cd -`		OK	OK	/home/vietdu91/42_works/minishell $>	0	Retourne à $OLD_PWD $OLD_PWD et $PWD sont inversés /home/vietdu91/42_works/minishell
	`$> cd --`	NON GERE	OK	OK	$>	0	Retourne à $HOME $OLD_PWD et $PWD sont inversés /home/vietdu91
	`$> cd ---`		OK	OK	bash: cd: --: invalid option $>	2	
	`$> cd $HOME`		OK	OK	$>	0	/home/vietdu91
	`$> cd $HOME $HOME`		OK	OK	bash: cd: too many arguments $>	1	
	`$> cd $HOME/42_works`		OK	OK	$>... unset: `/home/vietdu91/42_works':	0	/home/vietdu91/42_works
	`$> cd "$PWD/srcs"`		OK	OK	$>	0	/home/vietdu91/42_works/minishell/srcs

Commande	Géré			Bash	Sortie attendue	Code	Commentaire
`$> cd '$PWD/srcs'`			OK	OK	`bash: cd: $PWD/srcs: No such file or directory` `$>`	1	
`$> unset HOME` `$> cd $HOME`			OK	OK	`bash: cd: HOME not set` `$>`	1	

`$> unset HOME`
`$> export HOME=`
`$> cd`
			OK	OK	`$>`	0	`/home/vietdu91/42_works/minishell`

`$> unset HOME`
`$> export HOME`
`$> cd`
			OK	OK	`bash: cd: HOME not set`
					`$>`
						1	

`$> cd minishell Docs crashtest.c`			OK	OK	`bash: cd: too many arguments` `$>`	1	
`$>    cd / │ echo $?` `$> pwd`			OK	OK	`0` `$>`	0 0	`/home/vietdu91/42_works/minishell`
`$> cd ~`			OK	OK	`$>`	0	`/home/vietdu91`
`$> cd ~/`	NON GERE		OK	OK	`$>`	0	`/home/vietdu91`
`$> cd ~/ │ echo $?` `$> pwd`	NON GERE		OK	OK	`0` `$>`	0	`/home/vietdu91/42_works/minishell`
`$> cd *`	NON GERE		OK	OK	`$>`	0	**[BONUS]** S'il n'y a qu'un seul <u>dossier</u> et rien d'autre
`$> cd *`	NON GERE		OK	OK	`bash: cd: minishell: No such file or directory` `$>`	1	**[BONUS]** S'il n'y a qu'un seul <u>fichier</u> et rien d'autre
`$> cd *`	NON GERE		OK	OK	`bash: cd: too many arguments` `$>`	1	**[BONUS]** S'il y a au moins deux éléments

`$> mkdir a`
`$> mkdir a/b`
`$> cd a/b`
`$> rm -r ../../a`
`$> cd ..`
			OK	OK	`chdir: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory`
					`$>`
						1	

`$> mkdir a`
`$> mkdir a/b`
`$> cd a/b`
`$> rm -r ../../a`
`$> pwd`
			OK	OK	`pwd: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory`
					`$>`
						1	

`$> mkdir a`
`$> mkdir a/b`
`$> cd a/b`
`$> rm -r ../../a`
`$> echo $PWD`
`$> echo $OLDPWD`
			OK	OK	`/home/vietdu91/42_works/minishell/a/b`
					`$>`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0
						0	

`$> mkdir a`
`$> mkdir a/b`
`$> cd a/b`
`$> rm -r ../../a`
`$> cd`
`$> echo $PWD`
`$> echo $OLDPWD`
			OK	OK	`/home`
					`$>`
					`/home/vietdu91/42_works/minishell/a/b`
					`$>`
						0
						0	

`$> mkdir a`
`$> cd a`
`$> rm -r ../a`
`$> echo $PWD`
`$> echo $OLDPWD`
			OK	OK	`../../../home/vietdu91/42_works/minishell/a`
					`$>`
					`../../../home/vietdu91/42_works/minishell`
					`$>`
						0
						0	

`$> export CDPATH=/` `$> cd $HOME/..`	NON GERE		OK	OK	`$>`	0 0	A gerer ? `/home`
`$> export CDPATH=/` `$> cd home/vietdu91`	NON GERE		OK	OK	`/home/vietdu91` `$>`	0 0	A gerer ? `/home/vietdu91`
`$> export CDPATH=./` `$> cd .`	NON GERE		OK	OK	`$>`	0 0	A gerer ? `/home/vietdu91/42_works/minishell`
`$> export CDPATH=./` `$> cd ..`	NON GERE		OK	OK	`$>`	0 0	A gerer ? `/home/vietdu91/42_works`
`$> chmod 000 minishell` `$> ./minishell`			OK	OK	`bash: ./minishell : Permission denied` `$>`	126	
`$> ls hola`			OK	OK	`ls: cannot access 'hola': No such file or directory`	2	
`$> ./Makefile`			OK	OK	`bash: ./Makefile: Permission denied` `$>`	126	
`$> ./minishell`			OK	OK	`$>`	0	Relance le Minishell Il faut donc 2 fois exit pour exit

`$> env │ grep SHLVL`
`$> ./minishell`
`$> env │ grep SHLVL`
`$> exit`
`$> env │ grep SHLVL`
	NON GERE		OK	OK	`$>`	0
						0
						0
						0
						0	

`$> touch hola` `$> ./hola`			OK	OK	`bash: ./hola: Permission denied` `$>`	126	
`$> env│"wc" -l`			OK	OK	`66` `$>`	0	
`$> env│"wc "-l`			OK	OK	`bash: wc -l: command not found` `$>`	127	
`$> expr 1 + 1`			OK	OK	`2` `$>`	0	
`$> expr $? + $?`			OK	OK	`0` `$>`	0	Fait la somme de la valeur de $?

`$> env -i ./minishell`
`$> env`
			OK	OK	`PWD=/mnt/nfs/homes/emtran/Desktop/42_minishell`
					`SHLVL=1`
					`_=/usr/bin/env`
					`$>`
						0	

`$> env -i ./minishell`
`$> export`
			OK	OK	`export OLDPWD`
					`export PWD="/mnt/nfs/homes/emtran/Desktop/42_minishell"`
					`export SHLVL="1"`
					`$>`
						0	

`$> env -i ./minishell` `$> cd`			OK	OK	`bash: cd: HOME not set` `$>`	1	
`$> env -i ./minishell` `$> cd ~`	NON GERE		OK	OK	`$>`	0	`/home/vietdu91`
`$> exit`			OK	OK	`exit` `~/minishell $>`	0	

`$> exit exit`			OK	OK	`exit`
					`bash: exit: exit: numeric argument required`
					`~/minishell $>`
						2	

`$> exit hola`			OK	OK	`exit`
					`bash: exit: hola: numeric argument required`
					`~/minishell $>`
						2	

`$> exit hola que tal`			OK	OK	`exit`
					`bash: exit: hola: numeric argument required`
					`~/minishell $>`
						2	

`$> exit 42`			OK	OK	`exit` `~/minishell $>`	42	
`$> exit 000042`			OK	OK	`exit` `~/minishell $>`	42	
`$> exit 666`			OK	OK	`exit` `~/minishell $>`	154	

`$> exit 666 666`			OK	OK	`exit`
					`bash: exit: too many arguments`
					`$>`
						1	

`$> exit -666 666`			OK	OK	`exit`
					`bash: exit: too many arguments`
					`$>`
						1	

`$> exit hola 666`			OK	OK	`exit`
					`bash: exit: hola: numeric argument required`
					`~/minishell $>`
						2	

`$> exit 666 666 666 666`			OK	OK	`exit`
					`bash: exit: too many arguments`
					`$>`
						1	

`$> exit 666 hola 666`			OK	OK	`exit` `bash: exit: too many arguments`	1	

`$> exit hola 666 666`			OK	OK	`exit`
					`bash: exit: hola: numeric argument required`
					`~/minishell $>`
						2	

`$> exit 259`			OK	OK	`exit` `~/minishell $>`	3	
`$> exit -4`			OK	OK	`exit` `~/minishell $>`	252	
`$> exit -42`			OK	OK	`exit` `~/minishell $>`	214	
`$> exit -0000042`			OK	OK	`exit` `~/minishell $>`	214	
`$> exit -259`			OK	OK	`exit` `~/minishell $>`	253	
`$> exit -666`			OK	OK	`exit` `~/minishell $>`	102	
`$> exit +666`			OK	OK	`exit` `~/minishell $>`	154	
`$> exit 0`			OK	OK	`exit` `~/minishell $>`	0	
`$> exit +0`			OK	OK	`exit` `~/minishell $>`	0	
`$> exit -0`			OK	OK	`exit` `~/minishell $>`	0	
`$> exit +42`			OK	OK	`exit` `~/minishell $>`	42	

`$> exit -69 -96`			OK	OK	`exit`
					`bash: exit: too many arguments`
					`$>`
						1	

`$> exit --666`			OK	OK	`exit`
					`bash: exit: --666: numeric argument required`
					`~/minishell $>`
						2	

`$> exit ++++666`			OK	OK	`exit`
					`bash: exit: ++++666: numeric argument required`
					`~/minishell $>`
						2	

Command						Output	Code	Comment

`$> exit ++++++0`				OK	OK	`exit`
						`bash: exit: ++++++0: numeric argument required`
						`~/minishell $>`
							2	

`$> exit ------0`				OK	OK	`exit`
						`bash: exit: ------0: numeric argument required`
						`~/minishell $>`
							2	

`$> exit "666"`				OK	OK	`exit` `~/minishell $>`	154	
`$> exit '666'`				OK	OK	`exit` `~/minishell $>`	154	
`$> exit '-666'`				OK	OK	`exit` `~/minishell $>`	102	Marche aussi pour les double quotes
`$> exit '+666'`				OK	OK	`exit` `~/minishell $>`	154	Marche aussi pour les double quotes

`$> exit '----666'`				OK	OK	`exit`
						`bash: exit: ----666: numeric argument required`
						`~/minishell $>`
							2	Marche aussi pour les double quotes

`$> exit '++++666'`				OK	OK	`exit`
						`bash: exit: ++++666: numeric argument required`
						`~/minishell $>`
							2	Marche aussi pour les double quotes

`$> exit '6'66`				OK	OK	`exit` `~/minishell $>`	154	Marche aussi pour les double quotes
`$> exit '2'66'32'`				OK	OK	`exit` `~/minishell $>`	8	Marche aussi pour les double quotes

`$> exit "'666'"`				OK	OK	`exit`
						`bash: exit: '666': numeric argument required`
						`~/minishell $>`
							2	

`$> exit '"666"'`				OK	OK	`exit`
						`bash: exit: "666": numeric argument required`
						`~/minishell $>`
							2	

`$> exit '666'"666"666`				OK	OK	`exit` `~/minishell $>`	170	
`$> exit +'666'"666"666`				OK	OK	`exit` `~/minishell $>`	170	
`$> exit -'666'"666"666`				OK	OK	`exit` `~/minishell $>`	86	
`$> exit 9223372036854775807`				OK	OK	`exit` `~/minishell $>`	255	

`$> exit 9223372036854775808`				OK	OK	`exit`
						`bash: exit: 9223372036854775808: numeric argument required`
						`~/minishell $>`
							2	

`$> exit -9223372036854775808`				OK	OK	`exit` `~/minishell $>`	0	

`$> exit -9223372036854775809`				OK	OK	`exit`
						`bash: exit: -9223372036854775809: numeric argument required`
						`~/minishell $>`
							2	

`$> cat | cat | cat | ls`				OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  srcs  test.c`
						`$>`
							0	Apparaît la ligne ls
								Et 3 retours à la ligne avec la touche Entrée

`$> ls | exit`				OK	OK	`$>`	0	Ne fait rien Chaque commande d'un pipeline est exécutée dans son propre sous-shell, qui est un processus distinct
`$> ls | exit 42`				OK	OK	`$>`	42	Ne fait rien

`$> exit | ls`				OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  srcs  test.c`
						`$>`
							0	

`$> echo hola > bonjour` `$> exit | cat -e bonjour`				OK	OK	`hola$` `$>`	0	
`$> echo hola > bonjour` `$> cat -e bonjour | exit`				OK	OK	`$>`	0	
`$> echo | echo`				OK	OK	`$>`	0	
`$> echo hola | echo que tal`				OK	OK	`que tal` `$>`	0	
`$> pwd | echo hola`				OK	OK	`hola` `$>`	0	
`$> env | echo hola`				OK	OK	`hola` `$>`	0	
`$> echo oui | cat -e`				OK	OK	`oui$` `$>`	0	
`$> echo oui | echo non | echo hola | grep oui`				OK	OK	`$>`	0	
`$> echo oui | echo non | echo hola | grep non`				OK	OK	`$>`	0	
`$> echo oui | echo non | echo hola | grep hola`				OK	OK	`hola` `$>`	0	
`$> echo hola | cat -e | cat -e | cat -e`				OK	OK	`hola$$$` `$>`	0	
`$> cd .. | echo "hola"`				OK	OK	`hola` `$>`	0	/home/vietdu91/42_works/minishell
`$> cd / | echo "hola"`				OK	OK	`hola` `$>`	0	/home/vietdu91/42_works/minishell
`$> cd .. | pwd`				OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0	/home/vietdu91/42_works/minishell

`$> ifconfig | grep ":"`				OK	OK	`docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500`
						`    ether 02:42:27:cf:95:49  txqueuelen 0  (Ethernet)`
						`[...]`
						`$>`
							0	

`$> ifconfig | grep hola`				OK	OK	`$>`	1	
`$> whoami | grep $USER`				OK	OK	`vietdu91` `$>`	0	
`$> whoami | grep $USER > /tmp/bonjour` `$> cat /tmp/bonjour`				OK	OK	`vietdu91` `$>`	0	
`$> whoami \| cat -e \| cat -e > /tmp/bonjour` `$> cat /tmp/bonjour`				OK	OK	`vietdu91$$` `$>`	0	

`$> whereis ls | cat -e | cat -e > /tmp/bonjour`
`$> cat /tmp/bonjour`
				OK	OK	`ls: /usr/bin/ls /usr/share/man/man1/ls.1posix.gz`
						`/usr/share/man/man1/ls.1.gz$$`
						`$>`
							0	

`$> ls | hola`				OK	OK	`bash: hola: command not found` `$>`	127	
`$> ls | ls hola`				OK	OK	`ls: cannot access 'hola': No such file or directory` `$>`	2	
`$> ls | ls | hola`				OK	OK	`bash: hola: command not found` `$>`	127	

`$> ls | hola | ls`				OK	OK	`bash: hola: command not found`
						`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  srcs  test.c`
						`$>`
							0	

`$> ls | ls | hola | rev`				OK	OK	`bash: hola: command not found` `$>`	0	
`$> ls | ls | echo hola | rev`				OK	OK	`aloh` `$>`	0	

`$> ls -la | grep "."`				OK	OK	`total 16`
						`drwx------  2 vietdu91 [...] .`
						`drwxrwxrwt 51 root     [...] ..`
						`-rwxr-xr-x  8 vietdu91 [...] crashtest.c`
						`[...]`
						`$>`
							0	Sort la commande entière de ls -la

`$> ls -la | grep "'.'"`				OK	OK	`$>`	0	
`$> echo test.c | cat -e| cat -e| cat -e| cat -e| cat -e| cat -e| cat -e| cat -e|cat -e|cat -e|cat -e`				OK	OK	`test.c$$$$$$$$$$$` `$>`	0	

`$> ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls|ls`				OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
						`philosophers srcs test.c`
						`$>`
							0	

`$> echo hola | cat | cat | cat | cat | cat | grep hola`				OK	OK	`hola` `$>`	0	
`$> echo hola | cat`				OK	OK	`hola` `$>`	0	
`$> echo hola| cat`				OK	OK	`hola` `$>`	0	
`$> echo hola |cat`				OK	OK	`hola` `$>`	0	
`$> echo hola|cat`				OK	OK	`hola` `$>`	0	
`$> echo hola || cat`	**NON GERE**			OK	OK	`hola` `$>`	0	**[BONUS]**

`$> echo hola ||| cat`				OK	OK	`bash: syntax error near`
						`unexpected token '|'`
						`$>`
							2	

`$> ech|o hola | cat`				OK	OK	`bash: ech: command not found`
						`bash: o: command not found`
						`$>`
							127	

`$> cat Makefile | cat -e | cat -e`				OK	OK	`[...]`
						`$$`
						`.PHONY:   all clean fclean re$$`
						`$>`
							0	

`$> cat Makefile | grep srcs | cat -e`				OK	OK	`objs/%.o:         srcs/%.c$`
						`                @norminette srcs/*$`
						`$>`
							0	

`$> cat Makefile | grep srcs | grep srcs | cat -e`				OK	OK	`objs/%.o:         srcs/%.c$`
						`                @norminette srcs/*$`
						`$>`
							0	

`$> cat Makefile | grep pr | head -n 5 | cd file_not_exist`				OK	OK	`bash: cd: file_not_exist: No such file or directory` `$>`	1	

Commande					Sortie	Code	Remarque
`$> cat Makefile | grep pr | head -n 5 | hello`			OK	OK	`bash: hello: command not found` `$>`	127	
`$> export HOLA=bonjour | cat -e | cat -e`			OK	OK	`$>`	0	
`$> unset HOLA | cat -e`			OK	OK	`$>`	0	

`$> export HOLA | echo hola`
`$> env | grep PROUT`
			OK	OK	`hola`
					`$>`
					`$>`
						0	

`$> export | echo hola`			OK	OK	`hola` `$>`	0	
`$> sleep 3 | sleep 3`			OK	OK	`$>`	0	Le prompt apparait au bout de 3 secondes

`$> time sleep 3 | sleep 3`			OK	OK	`real0m3.002s`
					`user0m0.003s`
					`sys0m0.001s`
					`$>`
						0	Le prompt apparait au bout de 3 secondes

`$> sleep 3 | exit`			OK	OK	`$>`	0	Le prompt apparait au bout de 3 secondes
`$> exit | sleep 3`			OK	OK	`$>`	0	Le prompt apparait au bout de 3 secondes

`$> echo hola > a`
`$> >>b echo que tal`
`$> cat a | <b cat | cat > c | cat`
			OK	OK	`$>`	0	

`$> pwd && ls`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> pwd || ls`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0	**[BONUS]**
`$> echo hola || echo bonjour`	NON GERE		OK	OK	`hola` `$>`	0	**[BONUS]**

`$> echo hola && echo bonjour`	NON GERE		OK	OK	`hola`
					`bonjour`
					`$>`
						0	**[BONUS]**

`$> echo bonjour || echo hola`	NON GERE		OK	OK	`bonjour` `$>`	0	**[BONUS]**

`$> echo bonjour && echo hola`	NON GERE		OK	OK	`bonjour`
					`hola`
					`$>`
						0	**[BONUS]**

`$> echo -n bonjour && echo -n hola`	NON GERE		OK	OK	`bonjourhola$>`	0	**[BONUS]**

`$> pwd && ls && echo hola`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`hola`
					`$>`
						0	**[BONUS]**

`$> pwd || ls && echo hola`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell`
					`hola`
					`$>`
						0	**[BONUS]**

`$> pwd && ls || echo hola`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> pwd || ls || echo hola`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0	**[BONUS]**

`$> ls || export ""`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> export "" || ls`	NON GERE		OK	OK	`bash: export: '': not a valid identifier`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> ls && export ""`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`bash: export: '': not a valid identifier`
					`$>`
						1	**[BONUS]**

`$> export "" && ls`	NON GERE		OK	OK	`bash: export: '': not a valid identifier` `$>`	1	**[BONUS]**

`$> ls || ;`	NON GERE		OK	OK	`bash: syntax error near`
					`unexpected token ';'`
					`$>`
						2	**[BONUS]** Marche pour les metacharacters |&;()

`$> ; || ls`	NON GERE		OK	OK	`bash: syntax error near`
					`unexpected token ';'`
					`$>`
						2	**[BONUS]** Marche pour les metacharacters |&;()

`$> ls && ;`	NON GERE		OK	OK	`bash: syntax error near`
					`unexpected token ';'`
					`$>`
						2	**[BONUS]** Marche pour les metacharacters |&;()

`$> ; && ls`	NON GERE		OK	OK	`bash: syntax error near`
					`unexpected token ';'`
					`$>`
						2	**[BONUS]** Marche pour les metacharacters |&;()

`$> ls || <`	NON GERE		OK	OK	`bash: syntax error near unexpected token 'newline'` `$>`	2	**[BONUS]** Marche pour les metacharacters <>
`$> ls && <`	NON GERE		OK	OK	`bash: syntax error near unexpected token 'newline'` `$>`	2	**[BONUS]** Marche pour les metacharacters <>
`$> cat | echo || ls`	NON GERE		OK	OK	`$>`	0	**[BONUS]**

`$> cat | echo && ls`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> ls || cat | echo`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> ls && cat | echo`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> export "" && unset ""`	NON GERE		OK	OK	`bash: export: '': not a valid identifier`	1	**[BONUS]**

`$> (ls)`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> ( ( ls ) )`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> ( ( ) ls )`	NON GERE		OK	OK	`bash: syntax error near unexpected token ')'` `$>`	2	**[BONUS]**

`$> ls && (ls)`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`$>`
						0	**[BONUS]**

`$> (ls && pwd)`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

`$> ( ( ls&&pwd ) )`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

`$> ( ( ls ) &&pwd )`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

`$> (ls && ( ( pwd ) ) )`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers srcs test.c`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

`$> (ls && pwd) > hola`
`$> cat hola`
	NON GERE		OK	OK	`crashtest.c`
					`Docs`
					`Dockers`
					`Drawings`
					`Makefile`
					`minishell`
					`philosophers`
					`srcs`
					`test.c`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

`$> > hola ls && pwd`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0	**[BONUS]**
`$> > hola (ls && pwd)`	NON GERE		OK	OK	`bash: syntax error near unexpected token '('` `$>`	2	**[BONUS]**

`$> (> pwd)`
`$> ls`
	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers pwd srcs test.c`
					`$>`
						0	**[BONUS]**

`$> (< pwd)` `$> ls`	NON GERE		OK	OK	`$>`	0	**[BONUS]** Si pwd existe
`$> (< pwd)`	NON GERE		OK	OK	`bash: pwd: No such file or directory` `$>`	1	**[BONUS]** Si pwd n'existe pas
`$> ( ( ( ( ( pwd) ) ) )`	NON GERE		OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0	**[BONUS]**
`$> () pwd`	NON GERE		OK	OK	`bash: syntax error near unexpected token ')'` `$>`	2	**[BONUS]**
`$> > pwd (ls)`	NON GERE		OK	OK	`bash: syntax error near unexpected token '('` `$>`	2	**[BONUS]**

`$> (ls||pwd)&&(ls||pwd)`	NON GERE		OK	OK	`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers pwd srcs test.c`
					`crashtest.c Docs Dockers Drawings Makefile minishell`
					`philosophers pwd srcs test.c`
					`$>`
						0	**[BONUS]**

`$> (lss||pwd)&&(lss||pwd)`	NON GERE		OK	OK	`bash: lss: command not found`
					`/home/vietdu91/42_works/minishell`
					`bash: lss: command not found`
					`/home/vietdu91/42_works/minishell`
					`$>`
						0	**[BONUS]**

Command	Status					Output	Exit	Notes
`$> (lss&&pwd)&&(lss&&pwd)`	NON GERE			OK	OK	`bash: lss: command not found`	127	[BONUS]

`$> (ls && pwd | wc) > hola`
`$> cat hola`
	NON GERE			OK	OK	`crashtest.c`
						`Docs`
						`Dockers`
						`Drawings`
						`Makefile`
						`minishell`
						`philosophers`
						`srcs`
						`test.c`
						`      1      1     34`
						`$>`
							0
							0
								[BONUS]

`$> (ls && pwd | wc) > hola`
`$> (ls && pwd | wc) > hola`
`$> cat hola`
	NON GERE			OK	OK	`crashtest.c`
						`Docs`
						`Dockers`
						`Drawings`
						`Makefile`
						`minishell`
						`philosophers`
						`srcs`
						`test.c`
						`      1      1     34`
						`$>`
							0
							0
							0
								[BONUS]

`$> (ls && pwd | wc) >> hola`
`$> echo hey&&(ls && pwd | wc) > hola`
`$> cat hola`
	NON GERE			OK	OK	`hey`
						`$>`
						`crashtest.c`
						`Docs`
						`[...]`
						`srcs`
						`test.c`
						`      1      1     34`
						`crashtest.c`
						`Docs`
						`[...]`
						`srcs`
						`test.c`
						`      1      1     34`
						`$>`
							0
							0
							0
								[BONUS]

`$> (pwd | wc) < hola`	NON GERE			OK	OK	`      1      1     34` `$>`	0	[BONUS]

`$> (ls && pwd | wc) < hola`	NON GERE			OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  pwd  srcs  test.c`
						`      1      1     34`
						`$>`
							0	[BONUS]

`$> (ls -z || pwd | wc) < hola`	NON GERE			OK	OK	`ls: invalid option -- 'z'` `$>`	0	[BONUS]
`$> echo hey > hola` `$> (pwd | wc) < hola`	NON GERE			OK	OK	`      1      1     34` `$>`	0	[BONUS]

`$> echo hey > hola`
`$> (ls && pwd | wc) < hola`
	NON GERE			OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  pwd  srcs  test.c`
						`      1      1     34`
						`$>`
							0	[BONUS]

`$> echo hey > hola` `$> (ls -z || pwd | wc) < hola`	NON GERE			OK	OK	`ls: invalid option -- 'z'` `$>`	0	[BONUS]

`$> (ls -z || pwd && ls)`	NON GERE			OK	OK	`ls: invalid option -- 'z'`
						`/home/vietdu91/42_works/minishell`
						`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  pwd  srcs  test.c`
						`$>`
							0	[BONUS]

`$> ls || (cat Makefile|grep srcs) && (pwd|wc)`	NON GERE			OK	OK	`crashtest.c  Docs  Dockers  Drawings  Makefile  minishell`
						`philosophers  pwd  srcs  test.c`
						`      1      1     34`
						`$>`
							0	[BONUS]

`$> ls -z && (ls) && (pwd)`	NON GERE			OK	OK	`ls: invalid option -- 'z'` `$>`	130	[BONUS]

`$> (ls > Docs/hey && pwd) > hola`
`$> cat hola`
`$> cat Docs/hey`
	NON GERE			OK	OK	`/home/vietdu91/42_works/minishell`
						`$>`
						`crashtest.c`
						`Docs`
						`[...]`
						`srcs`
						`test.c`
						`$>`
							0
							0
							0
								[BONUS]

`$> ls > Docs/hey && pwd > hola`
`$> cat hola`
`$> cat Docs/hey`
	NON GERE			OK	OK	`/home/vietdu91/42_works/minishell`
						`$>`
						`crashtest.c`
						`Docs`
						`[...]`
						`srcs`
						`test.c`
						`$>`
							0
							0
							0
								[BONUS]

`$> cd ../.. && pwd && pwd`	NON GERE			OK	OK	`/home/vietdu91/`
						`/home/vietdu91/`
						`$>`
							0	[BONUS]

`$> (cd ../.. && pwd) && pwd`	NON GERE			OK	OK	`/home/vietdu91/`
						`/home/vietdu91/42_works/minishell`
						`$>`
							0	[BONUS]

`$> ls -z || cd ../../..&&pwd`
`$> pwd`
	NON GERE			OK	OK	`ls: invalid option -- 'z'`
						`/home/vietdu91`
						`$>`
						`/home/vietdu91`
						`$>`
							0
							0
								[BONUS]

`$> ls -z || (cd ../../..&&pwd)`
`$> pwd`
	NON GERE			OK	OK	`ls: invalid option -- 'z'`
						`/home/vietdu91`
						`$>`
						`/home/vietdu91/42_works/minishell`
						`$>`
							0
							0
								[BONUS]

`$> echo hola > bonjour` `$> cat bonjour`				OK	OK	`hola` `$>`	0 0	

`$> echo que tal >> bonjour`
`$> cat bonjour`
				OK	OK	`hola`
						`que tal`
						`$>`
							0
							0	

`$> echo hola > bonjour`
`$> echo que tal >> bonjour`
`$> cat < bonjour`
				OK	OK	`$>`
						`hola`
						`que tal`
						`$>`
							0
							0
							0	

`$> echo hola > bonjour`
`$> rm bonjour`
`$> echo que tal >> bonjour`
`$> cat < bonjour`
				OK	OK	`que tal`
						`$>`
							0
							0
							0
							0	

`$> echo hola que tal > bonjour` `$> cat bonjour`				OK	OK	`hola que tal` `$>`	0 0	
`$> echo hola que tal > /tmp/bonjour` `$> cat -e /tmp/bonjour`				OK	OK	`hola que tal$` `$>`	0 0	

`$> export HOLA=hey`
`$> echo bonjour > $HOLA`
`$> echo $HOLA`
				OK	OK	`$>`
						`hola`
						`$>`
							0
							0
							0
								Ne fait rien

`$> whereis grep > Docs/bonjour`
`$> cat Docs/bonjour`
				OK	OK	`grep: /usr/bin/grep /usr/share/man/man1/grep.1.gz`
						`/usr/share/man/man1/grep.1posix.gz /usr/share/info/grep.`
						`info.gz`
						`$>`
							0
							0	

`$> ls -la > Docs/bonjour`
`$> cat Docs/bonjour`
				OK	OK	`total 16`
						`drwx------  2 vietdu91 [...] .`
						`drwxrwxewt 51 root     [...] ..`
						`-rwxr-xr-x  8 vietdu91 [...] crashtest.c`
						`[...]`
						`$>`
							0
							0	

`$> pwd>bonjour` `$> cat bonjour`				OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0 0	
`$> pwd >              bonjour` `$> cat bonjour`				OK	OK	`/home/vietdu91/42_works/minishell` `$>`	0 0	
`$> echo hola > > bonjour`				OK	OK	`bash: syntax error near unexpected token '>'` `$>`	2	
`$> echo hola < < bonjour`				OK	OK	`bash: syntax error near unexpected token '<'` `$>`	2	
`$> echo hola >>> bonjour`				OK	OK	`bash: syntax error near unexpected token '>'` `$>`	2	
`$> > bonjour echo hola` `$> cat bonjour`				OK	OK	`hola` `$>`	0 0	

`$> > bonjour | echo hola`
`$> cat bonjour`
				OK	OK	`hola`
						`$>`
						`$>`
							0
							0	

`$> prout hola > bonjour`
`$> ls`
				OK	OK	`bash: prout: command not found`
						`$>`
						`bonjour [...]`
							127	Erreur mais file "bonjour" cree tout de meme

`$> echo hola > hello >> hello >> hello`
`$> ls`
`$> cat hello`
				OK	OK	`$>`
						`hello`
						`$>`
						`hola`
						`$>`
							0
							0
							0	

`$> echo hola > hello >> hello >> hello`
`$> echo hola >> hello`
`$> cat < hello`
				OK	OK	`hola`
						`hola`
						`$>`
							0
							0
							0	

`$> echo hola > hello >> hello >> hello`
`$> echo hola >> hello`
`$> echo hola > hello >> hello >> hello`
`$> cat < hello`
				OK	OK	`hola`
						`$>`
							0
							0
							0
							0	

`$> echo hola >> hello >> hello > hello`
`$> echo hola >> hello`
`$> cat < hello`
				OK	OK	`hola`
						`hola`
						`$>`
							0
							0
							0	

`$> echo hola >> hello >> hello > hello`
`$> echo hola >> hello`
`$> echo hola >> hello >> hello > hello`
`$> cat < hello`
				OK	OK	`hola`
						`$>`
							0
							0
							0
							0	

Commande				OK	OK	Sortie	Code	Commentaire

`$> echo hola > hello`
`$> echo hola >> hello >> hello >> hello`
`$> echo hola >> hello`
`$> cat < hello`
				OK	OK	hola
						hola
						hola
						$>
							0
							0
							0
							0	

`$> echo hola > hello`
`$> echo hey > bonjour`
`$> echo <bonjour <hello`
				OK	OK	$>	0
							0
							0	

`$> echo hola > hello`
`$> echo hey > bonjour`
`$> echo <hello <bonjour`
				OK	OK	$>	0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> rm bonjour hello`
`$> echo hola > bonjour > hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> echo hola > bonjour > hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> rm bonjour hello`
`$> echo hola > bonjour >> hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> echo hola > bonjour > hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> rm bonjour hello`
`$> echo hola > bonjour > hello >> bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> echo hola > bonjour > hello >> bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> rm bonjour hello`
`$> echo hola >> bonjour > hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> echo hola >> bonjour > hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> rm bonjour hello`
`$> echo hola >> bonjour >> hello >> bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						$>
						$>
							0
							0
							0
							0
							0
							0	

`$> echo hola > bonjour`
`$> echo hey > hello`
`$> echo hola >> bonjour >> hello > bonjour`
`$> cat bonjour`
`$> cat hello`
				OK	OK	hola
						hola
						$>
						hey
						$>
							0
							0
							0
							0
							0	

`$> > bonjour echo hola bonjour` `$> cat bonjour`				OK	OK	bonjour hola $>	0 0	

`$> >bonjour echo >`
`hola>bonjour>hola>>bonjour>hola hey >bonjour`
`hola >hola`
`$> cat bonjour`
`$> cat hola`
				OK	OK	$>
						hey hola
						$>
							0
							0
							0	

`$> echo bonjour > hola1`
`$> echo hello > hola2`
`$> echo 2 >hola1 >> hola2`
`$> ls`
`$> cat hola1`
`$> cat hola2`
				OK	OK	hola1 hola2 [...]
						$>
						$>
						hello
						2
						$>
							0
							0
							0
							0
							0
							0	

`$> echo bonjour > hola1`
`$> echo hello > hola2`
`$> echo 2 >>hola1 > hola2`
`$> ls`
`$> cat hola1`
`$> cat hola2`
				OK	OK	hola1 hola2 [...]
						$>
						bonjour
						$>
						2
						$>
							0
							0
							0
							0
							0
							0	

`$> > pwd` `$> ls`				OK	OK	pwd $>	0 0	
`$> < pwd`				OK	OK	bash: pwd: No such file or directory $>	1	
`$> < Makefile .`		NON GERE		OK	OK	.: not enough arguments $>	1	Voir pourquoi
`$> cat <pwd`				OK	OK	bash: pwd: No such file or directory $>	1	
`$> cat <srcs/pwd`				OK	OK	bash: srcs/pwd: No such file or directory $>	1	
`$> cat <../pwd`				OK	OK	bash: ../pwd: No such file or directory $>	1	
`$> cat >>`				OK	OK	bash: syntax error near unexpected token `newline' $>	2	Marche aussi pour les autres operateurs de redirection
`$> cat >>>`				OK	OK	bash: syntax error near unexpected token `>' $>	2	
`$> cat >> <<`				OK	OK	bash: syntax error near unexpected token `<<' $>	2	
`$> cat >> > >> << >>`				OK	OK	bash: syntax error near unexpected token `>' $>	2	
`$> cat < ls`				OK	OK	bash: ls: No such file or directory $>	1	
`$> cat < ls > ls`				OK	OK	bash: ls: No such file or directory $>	1	
`$> cat > ls1 < ls2` `$> ls`				OK	OK	bash: ls2: No such file or directory $> ls1 [...] $>	1 0	
`$> >>hola` `$> cat hola`				OK	OK	$>	0 0	
`$> echo hola > bonjour` `$> cat < bonjour`				OK	OK	hola $>	0 0	
`$> echo hola >bonjour` `$> cat <bonjour`				OK	OK	hola $>	0 0	
`$> echo hola>bonjour` `$> cat<bonjour`				OK	OK	hola $>	0 0	
`$> echo hola> bonjour` `$> cat< bonjour`				OK	OK	hola $>	0 0	
`$> echo hola          >bonjour` `$> cat<          bonjour`				OK	OK	hola $>	0 0	
`$> echo hola      >      bonjour` `$> cat      <      bonjour`				OK	OK	hola $>	0 0	
`$> echo hola > srcs/bonjour` `$> cat < srcs/bonjour`				OK	OK	hola $>	0 0	
`$> echo hola >srcs/bonjour` `$> cat <srcs/bonjour`				OK	OK	hola $>	0 0	

`$> echo hola > bonjour`
`$> echo que tal >> bonjour`
`$> cat < bonjour`
				OK	OK	hola
						que tal
						$>
							0
							0
							0	

`$> echo hola > bonjour`
`$> rm bonjour`
`$> echo que tal >> bonjour`
`$> cat < bonjour`
				OK	OK	que tal
						$>
							0
							0
							0
							0	

`$> e'c'"h"o hola > bonjour` `$> cat 'bo'"n"jour`				OK	OK	hola $>	0 0	

`$> echo hola > bonjour\ 1`
`$> ls`
`$> cat bonjour\ 1`
		NON GERE		OK	OK	bonjour 1' [...]
						$>
						hola
						$>
							0
							0
							0	

`$> echo hola > bonjour hey`
`$> ls`
`$> cat bonjour`
`$> cat hey`
				OK	OK	bonjour [...]
						$>
						hola hey
						$>
						cat: hey: No such file or directory
						$>
							0
							0
							0
							1	

`$> echo hola > srcs/bonjour`
`$> echo hey > srcs/hello`
`$> >srcs/bonjour >srcs/hello <prout`
`$> cat srcs/bonjour srcs/hello`
				OK	OK	bash: prout: No such file or directory
						$>
						$>
						$>
							0
							0
							1
							0	

`$> echo hola > srcs/bonjour`
`$> echo hey > srcs/hello`
`$> rm srcs/bonjour srcs/hello`
`$> >srcs/bonjour >srcs/hello <prout`
`$> ls srcs`
`$> cat srcs/bonjour srcs/hello`
				OK	OK	bash: prout: No such file or directory
						$>
						bonjour hello
						$>
						$>
							0
							0
							0
							1
							0
							0	

`$> echo hola > srcs/bonjour`
`$> echo hey > srcs/hello`
`$> >srcs/bonjour <prout >srcs/hello`
`$> cat srcs/bonjour`
`$> cat srcs/hello`
				OK	OK	bash: prout: No such file or directory
						$>
						$>
						hey
						$>
							0
							0
							1
							0
							0	

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> rm srcs/bonjour srcs/hello
$> >srcs/bonjour <prout >srcs/hello
$> ls srcs
$> cat srcs/bonjour
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
```
0 0 0 1 0 0

```
$> echo hola > ../bonjour
$> echo hey > ../hello
$> >../bonjour >../hello <prout
$> cat ../bonjour ../hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
```
0 0 1 0

```
$> echo hola > ../bonjour
$> echo hey > ../hello
$> rm ../bonjour ../hello
$> >../bonjour >../hello <prout
$> ls ..
$> cat ../bonjour ../hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour   hello
$>
$>
```
0 0 0 1 0 0

```
$> echo hola > ../bonjour
$> echo hey > ../hello
$> >../bonjour <prout >../hello
$> cat ../bonjour
$> cat ../hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > ../bonjour
$> echo hey > ../hello
$> rm ../bonjour ../hello
$> >../bonjour <prout >../hello
$> ls ..
$> cat ../bonjour
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
$>
```
0 0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >srcs/bonjour >>srcs/hello <prout
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >>srcs/bonjour >srcs/hello <prout
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >>srcs/bonjour >>srcs/hello <prout
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >srcs/bonjour <prout >>srcs/hello
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >>srcs/bonjour <prout >srcs/hello
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> >>srcs/bonjour <prout >>srcs/hello
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > srcs/bonjour
$> echo hey > srcs/hello
$> <prout >>srcs/bonjour >>srcs/hello
$> cat srcs/bonjour
$> cat srcs/hello
```
OK | OK
```
bash: prout: No such file or directory
$>
bonjour
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > bonjour
$> echo hey > hello
$> <bonjour >hello
$> cat bonjour
$> cat hello
```
OK | OK
```
hola
$>
>>
```
0 0 0 0 0

```
$> echo hola > bonjour
$> echo hey > hello
$> >bonjour >hello < prout
$> cat bonjour
$> cat hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
$>
$>
```
0 0 1 0 0

```
$> echo hola > bonjour
$> echo hey > hello
$> rm bonjour hello
$> >bonjour >hello < prout
$> cat bonjour
$> cat hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
$>
$>
```
0 0 0 1 0 0

```
$> echo hola > bonjour
$> echo hey > hello
$> >bonjour <prout hello
$> cat bonjour
$> cat hello
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
hey
$>
```
0 0 1 0 0

```
$> echo hola > bonjour
$> echo hey > hello
$> rm bonjour hello
$> >bonjour <prout hello
$> cat bonjour
```
OK | OK
```
bash: prout: No such file or directory
$>
$>
$>
```
Le fichier hello n'existe pas
0 0 0 1 0

```
$> echo hola > bonjour
$> <bonjour cat | wc > bonjour
$> cat bonjour
```
OK | OK
```
       0       0       0
$>
```
0 0 0

```
$> rm -f bonjour
$> rm bonjour > bonjour
$> ls -l bonjour
```
OK | OK
```
ls: cannot access 'bonjour': No such file or directory
```
0 0 2

```
$> export HOLA="bonjour hello"
$> >$HOLA
$> ls
```
OK | OK
```
'bonjour hello' [...]
$>
```
0 0 0

```
$> export HOLA="bonjour hello"
$> >"$HOLA"
$> ls
```
OK | OK
```
'bonjour hello' [...]
$>
```
0 0 0

```
$> export HOLA="bonjour hello"
$> >$"HOLA"
$> ls
```
OK | OK
```
HOLA [...]
$>
```
0 0 0

```
$> export HOLA="bonjour hello"
$> >$HOLA>hey
$> ls
```
OK | OK
```
'bonjour hello' hey [...]
$>
```
0 0 0

```
$> export HOLA="bonjour hello"
$> >hey>$HOLA
$> ls
```
OK | OK
```
'bonjour hello' hey [...]
$>
```
0 0 0

```
$> export HOLA="bonjour hello"
$> >hey>$HOLA>hey
$> ls
```
OK | OK
```
'bonjour hello' hey [...]
$>
```
0 0 0

```
$> export A=hey
$> export A B=Hola D E C="Que Tal"
$> echo $PROUT$B$C > /tmp/a > /tmp/b > /tmp/c
$> cat /tmp/a
$> cat /tmp/b
$> cat /tmp/c
```
OK | OK
```
$>
$>
$>
HolaQue Tal
$>
```
0 0 0 0 0 0

```
!!!!! Contenu du fichier a : "Amour Tu es Horrible"
!!!!! Contenu du fichier b : "0123456789"
!!!!! Contenu du fichier c : "Prout"
```

```
$> <a cat <b <c
```
OK | OK
```
Prout
$>
```
0

```
$> <a cat <b <c
$> cat a
$> cat b
$> cat c
```
OK | OK
```
$>
$>
$>
Amour Tu es Horrible
0123456789
Prout
$>
```
0 0 0 0

```
$> >a ls >b >>c >d
$> cat a
$> cat b
$> cat c
$> cat d
```
OK | OK
```
$>
$>
$>
Prout
crashtest.c
Docs
[...]
test.c
$>
```
0 0 0 0 0

```
$> >a ls >b >>c >d
$> cat a
$> cat b
$> cat c
$> cat d
```
OK | OK
```
$>
$>
$>
Prout
$>
crashtest.c
Docs
[...]
test.c
$>
```
0 0 0 0 0

```
$> echo hola > a >.b > .c.
$> cat a
$> cat b
$> cat c
```
OK | OK
```
$>
$>
hola
$>
```
0 0 0 0

```
$> mkdir dir
$> ls -la > dir/bonjour
$> cat dir/bonjour
```
OK | OK
```
total 16
drwx------  2 vietdu91 [...] .
drwxrwxewt 51 root     [...] ..
[...]
$>
```
0 0 0

```
$> <a
$> cat a
```
OK | OK
```
Amour Tu es Horrible
$>
```
0 0

```
$> >d cat <a >>e
$> cat a
$> cat d
$> cat e
```
OK | OK
```
$>
Amour
$>
Amour
$>
```
0 0 0 0

```
$> < a > b cat > hey >> d
$> cat d
$> ls
```
OK | OK
```
Amour Tu Es Horrible
$>
[...] a b hey d
$>
```
0 0 0