

MINISHELL

ATTENTION

Les comportements presents dans cet Excel ne sont pas tous a gerer dans la Mandatory du projet Minishell !

Ne vous mettez pas la pression pour essayer de tous les gerer !

Verifiez juste pour ces cas non gerables s'ils ne Segfault pas, ou n'effectuent pas un comportement suspect !

Utilisez votre bon sens pour exploiter au mieux ce document et faire un Minishell pas trop ambitieux, selon votre situation du projet.

(Par contre, si vous etes chauds pour tout gerer, GG a vous !)

Bon courage et n'hesitez pas a m'envoyer un MP si besoin !

(GitHub : [vietdu91](#) / Discord : [vietdu91#6048](#) / Intra : [emtran](#))

Gestion des comportements

[illegible]

ECHO	\$> ..	NON GERE		OK	OK	bash: ...: command not found	127		
	\$> ~	NON GERE		OK	OK	bash: /home/vietdu91: Is a directory	126		
	\$> ABC=holà	NON GERE		OK	OK	\$>	0	/!\ A gerer comme une erreur	
	\$> 4ABC=holà	NON GERE		OK	OK	bash: 4ABC=holà: command not found	127		
	holà			OK	OK	bash: holà: command not found	127		
	\$> holà que tal			OK	OK	bash: holà: command not found	127		
	\$> Makefile			OK	OK	bash: Makefile: command not found	127		
	\$> echo			OK	OK	\$>	0		
	\$> echo -n			OK	OK	\$>	0		
	\$> echo Holà			OK	OK	Holà	0		
	\$> echoHolà			OK	OK	bash: echoHolà: command not found	127		
	\$> echo-nHolà			OK	OK	bash: echo-nHolà: command not found	127		
	\$> echo -n Holà			OK	OK	Holà\$>	0		
	\$> echo "-n" Holà			OK	OK	Holà\$>	0		
	\$> echo -nHolà			OK	OK	-nHolà	0		
	\$> echo Holà -n			OK	OK	Holà -n	0		
	\$> echo Holà Que Tal			OK	OK	Holà Que Tal	0		
	\$> echo Holà			OK	OK	Holà	0		
	\$> echo Holà Que Tal			OK	OK	Holà Que Tal	0		
	\$> echo \n holà			OK	OK	\n holà	0	A gerer ?	
	\$> echo " " cat -e			OK	OK	\$	0		
	\$> echo cat -e			OK	OK	\$	0		
	\$> ""echo holà"""" que"""" tal""""			OK	OK	holà que tal	0		
	\$> echo -n -n			OK	OK	\$>	0		
	\$> echo -n -n Holà Que			OK	OK	Holà Que\$>	0		
	\$> echo -p			OK	OK	-p	0		
	\$> echo -nnnnn			OK	OK	\$>	0		
	\$> echo -n -nnn -nnnn			OK	OK	\$>	0		
	\$> echo -n -nn -nnnn			OK	OK	-n -nn -nnnn	0		
	\$> echo -n -nnn holà -nnnn			OK	OK	holà -nnnn\$>	0		
	\$> echo -n -nnn -nnnn			OK	OK	-nnn -nnnn\$>	0		
	\$> echo -nnnn -nn			OK	OK	nnnn -nn	0		
	\$> echo -nnn -nnnn -nn			OK	OK	nnnn -nn\$>	0		
	\$> echo -nnn -nnnn -nnnn			OK	OK	nnnn -nn -nnnn\$>	0		
	\$> echo -nnn -nnnn -nnnn			OK	OK	nnnn -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 \$1\$			OK	OK	0\$	0		
	\$> echo \$? echo \$? echo \$?			OK	OK	0	0		
	\$> echo \$1\$= cat -e			OK	OK	\$1\$=\$	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 \$UID\$HOME			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 \$USERSvar[\$USER\$USER]\$USERTest\$USER			OK	OK	vietdu91\$USERvietdu91\$USERTestvietdu91	0	A ne pas gerer si '\' non gere : Doit sortir : vietdu91vietdu91vietdu91vietdu91	
	\$> echo \$holà*			OK	OK	\$>	0	Cas d'un variable inexistant	
	\$> echo -nnnn \$holà			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 holà			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 holà	NON GERE		OK	OK	crashtest.c test.c holà	0	[BONUS]	
	\$> echo *t	NON GERE		OK	OK	crashtest.c test.c	0	[BONUS]	
	\$> echo \$*	NON GERE		OK	OK	\$>	0	[BONUS]	
	\$> echo holà*holà *	NON GERE		OK	OK	holà*holà crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c	0	[BONUS]	
	\$> echo \$holà*	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		

x EXPORT & UNSET	\$ sleep 3 sleep 3 sleep 3 (faire Ctrl-\ 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		
	\$ 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			OK	OK	... HOLA=bonjour	0		
	\$ env			OK	OK	... HOLA=bonjour	0		
	\$ export HOLA=bonjour			OK	OK	... HOLA=bonjour	0		
	\$ export			OK	OK	... export PATH=..."	0		Les variables sont trieés par ordre alphabetique On n'oublie pas les doubles quotes qui designent les strings attribues a chaque variable
	\$ export Hola			OK	OK	... export HOME="/home/vietdu91"	0		Hola n'existe pas dans env mais il existe dans export
	\$ export			OK	OK	export Hola export PATH=..."	0		
	\$ export Hola9hey			OK	OK	... export HOME="/home/vietdu91"	0		
	\$ export			OK	OK	export Hola9hey export PATH=..."	0		
	\$ export SDOCONTEXTIST			OK	OK	... export HOME="/home/vietdu91"	0		Le cas ou la variable n'existe pas et n'est pas inportee
	\$ export grep "HOME"			OK	OK	export PATH=..."	0		
	\$ export ""			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 7=2			OK	OK	bash: export: '7=2': not a valid identifier	1		
	\$ export 9HOLA=			OK	OK	bash: export: '9HOLA=': not a valid identifier	1		
	\$ export HOLA=bonjour			OK	OK	\$>	0		
	\$ env			OK	OK	HOLA=bonjour	0		
	\$ export _HOLA=bonjour			OK	OK	... _HOLA=bonjour	0		
	\$ env			OK	OK	... _HOLA=bonjour	0		
	\$ export __HOLA=bonjour			OK	OK	... __HOLA=bonjour	0		
	\$ env			OK	OK	... __HOLA=bonjour	0		
	\$ export _H0_LA=bonjour			OK	OK	_H0_LA=bonjour	0		
	\$ env			OK	OK	_H0_LA=bonjour	0		
	\$ export HOL@=bonjour			OK	OK	bash: export: 'HOL@=bonjour': not a valid identifier	1		Cas où le non 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 H0-LA=bonjour			OK	OK	bash: export: 'H0-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 H0\\A=bonjour	NON GERE		OK	OK	bash: export: 'H0\\A=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 H0*L=bonjour			OK	OK	bash: export: 'H0*L=bonjour': not a valid identifier	1		
	\$ export H0WLA=bonjour			OK	OK	bash: export: 'H0WLA=bonjour': not a valid identifier	1		
	\$ export HQ@LA=bonjour			OK	OK	bash: export: 'HQ@LA=bonjour': not a valid identifier	1		
	\$ export H0!LA=bonjour			OK	OK	bash: !LA=bonjour: event not found	0		
	\$ export H0\$?LA=bonjour			OK	OK	... HQ@LA=bonjour	0		
	\$ env			OK	OK	HQ@LA=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	NON GERE		OK	OK	... HOLA=bonjour	0		
	\$ env			OK	OK	... HOLA=bonjour	0		
	\$ export HOLA=bonjour	NON GERE		OK	OK	... HOLA=bonjourbonjour	0		"+=" s'ajoute au contenu de la variable
	\$ export HOLA=bonjour	NON GERE		OK	OK	\$>	0		
	\$ env			OK	OK	... PATH=...	0		
	\$ export HOLA=bonjour	NON GERE		OK	OK	\$>	0		
	\$ env			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			OK	OK	... HOLA=bon	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA= bonjour			OK	OK	... HOLA=	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=bonsotr			OK	OK	... HOLA=bonjour	0		
	\$ export HOLA=bonretour			OK	OK	\$>	0		
	\$ export HOLA=bonjour			OK	OK	... HOLA=bonjour	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=\$HOME			OK	OK	... HOLA=/home/vietdu91	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=bonjour\$HOME			OK	OK	... HOLA=bonjour/home/vietdu91	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=\$HOMEbonjour			OK	OK	... HOLA=	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=bon\$jour			OK	OK	... HOLA=bon	0		Le cas où la variable "\$jour" n'existe pas
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=bonjour	NON GERE		OK	OK	... HOLA=bonjour	0		
	\$ env			OK	OK	\$>	0		
	\$ export HOLA=bonjour	NON GERE		OK	OK	... HOLA=bonjour	0		
	\$ env			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		

	\$> unset 9HOLA			OK	OK	bash: unset: '9HOLA': not a valid identifier	1		
	\$> unset HOLA9			OK	OK	\$>	0		
	\$> unset HOL7A			OK	OK	bash: unset: 'HOL7A': not a valid identifier	1		
	\$> unset HOLA HOL7A			OK	OK	bash: unset: 'HOL7A': not a valid identifier	1		Il unset bien HOLA
	\$> unset HOL7A HOLA			OK	OK	bash: unset: 'HOL7A': not a valid identifier	1		Il unset bien HOLA
	\$> unset HOL7A HOL_A			OK	OK	bash: unset: 'HOL7A': 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		
	\$> 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\$7A			OK	OK	\$>	0		Unset prend en compte le contenu de \$?, c'est-a-dire qu'il lit HOLA ou HOL127A
	\$> unset HOL~A			OK	OK	bash: unset: 'HOL~': not a valid identifier	1		
	\$> unset "" HOLA			OK	OK	bash: unset: '': not a valid identifier	1		HOLA sera quand mène unset !!
	\$> env grep HOLA			OK	OK	\$>	0		
	\$> unset PATH			OK	OK	\$>	0		
	\$> echo \$PATH			OK	OK	\$>	0		
	\$> unset PATH			OK	OK	bash: cat: No such file or directory	0		
	\$> cat /etc/passwd			OK	OK	\$>	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		
	\$> 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			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			OK	OK	\$>	0		On doit se retrouver dans le dossier défini par \$HOME
	\$> cd .			OK	OK	/home/vietdu91	0		
	\$> cd ./			OK	OK	\$>	0		L'utilisateur reste dans le même dossier
	\$> cd ./././			OK	OK	/home/vietdu91/42_works/minishell	0		
	\$> cd ././././			OK	OK	/home/vietdu91/42_works/minishell	0		
	\$> cd ..			OK	OK	\$>	0		On se retrouve au dossier précédent
	\$> cd ../			OK	OK	/home/vietdu91/42_works	0		
	\$> cd ../.			OK	OK	/home/vietdu91	0		
	\$> cd ../.			OK	OK	/home/vietdu91/42_works	0		
	\$> cd .././././			OK	OK	/home/vietdu91/42_works	0		
	\$> cd srcs			OK	OK	/home/vietdu91/42_works/minishell/srcs	0		
	\$> cd srcs objs			OK	OK	/home/vietdu91/42_works/minishell	1		
	\$> cd "srcs"			OK	OK	\$>	0		/home/vietdu91/42_works/minishell/srcs
	\$> cd "/etc"			OK	OK	\$>	0		/home/vietdu91/42_works/minishell/srcs
	\$> cd /e"tc"			OK	OK	\$>	0		Marche aussi avec les double quotes
	\$> cd /e"tc"			OK	OK	\$>	0		/
	\$> 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	\$> cd //	NON GERE		OK	OK	//	0		/
	\$> pwd	NON GERE		OK	OK	\$>	0		Marche aussi avec les double quotes
	\$> cd '/'			OK	OK	/	0		/
	\$> pwd			OK	OK	\$>	0		/
	\$> cd //			OK	OK	/	0		/
	\$> pwd			OK	OK	\$>	0		/
	\$> cd /////////			OK	OK	/	0		/
	\$> pwd			OK	OK	\$>	0		Marche aussi avec les double quotes
	\$> cd '////////'			OK	OK	/	0		/
	\$> pwd			OK	OK	\$>	1		
	\$> cd /minishell			OK	OK	bash: cd: /minishell: No such file or directory	0		
	\$> 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	\$>	0		/home/vietdu91/42_works
	\$> cd "\$PWD/srcs"			OK	OK	\$>	0		/home/vietdu91/42_works/minishell/srcs
	\$> cd '\$PWD/srcs'			OK	OK	bash: cd: \$PWD/srcs: No such file or directory	1		
	\$> unset HOME			OK	OK	bash: cd: HOME not set	1		
	\$> cd \$HOME			OK	OK	\$>	0		/home/vietdu91/42_works/minishell
	\$> unset HOME			OK	OK	\$>	0		
	\$> export HOME=			OK	OK	\$>	0		
	\$> cd			OK	OK	\$>	0		
	\$> unset HOME			OK	OK	bash: cd: HOME not set	1		
	\$> export HOME			OK	OK	\$>	0		
	\$> cd			OK	OK	\$>	0		
	\$> cd minishell Docs crashtest.c			OK	OK	bash: cd: too many arguments	1		
	\$> cd / echo \$?			OK	OK	0	0		/home/vietdu91/42_works/minishell
	\$> pwd			OK	OK	\$>	0		
	\$> cd -			OK	OK	\$>	0		/home/vietdu91
	\$> cd -/	NON GERE		OK	OK	\$>	0		/home/vietdu91
	\$> cd -/ echo \$?	NON GERE		OK	OK	0	0		/home/vietdu91/42_works/minishell
	\$> pwd	NON GERE		OK	OK	\$>	0		
	\$> 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			OK	OK	chdir: error retrieving current directory: getcwd: cannot	1		
	\$> mkdir a/b			OK	OK	access parent directories: No such file or directory	\$>		
	\$> cd a/b			OK	OK	\$>	0		
	\$> rm -r .././a			OK	OK	\$>	0		
	\$> cd ..			OK	OK	\$>	0		
	\$> mkdir a			OK	OK	pwd: error retrieving current directory: getcwd: cannot	1		
	\$> mkdir a/b			OK	OK	access parent directories: No such file or directory	\$>		
	\$> cd a/b			OK	OK	\$>	0		
	\$> rm -r .././a			OK	OK	\$>	0		
	\$> pwd			OK	OK	\$>	0		
	\$> mkdir a			OK	OK	/home/vietdu91/42_works/minishell/a/b	0		
	\$> mkdir a/b			OK	OK	/home/vietdu91/42_works/minishell	\$>	0	
	\$> cd a/b			OK	OK	\$>	0		
	\$> rm -r .././a			OK	OK	/home/vietdu91/42_works/minishell/a/b	\$>	0	
	\$> echo \$PWD			OK	OK	\$>	0		
	\$> echo \$OLDPWD			OK	OK	\$>	0		
	\$> mkdir a			OK	OK	/home	\$>	0	
	\$> mkdir a/b			OK	OK	/home	\$>	0	
	\$> cd a/b			OK	OK	/home/vietdu91/42_works/minishell/a/b	\$>	0	
	\$> rm -r .././a			OK	OK	\$>	0		
	\$> cd			OK	OK	\$>	0		
	\$> echo \$PWD			OK	OK	.././../home/vietdu91/42_works/minishell/a	\$>	0	
	\$> echo \$OLDPWD			OK	OK	.././../home/vietdu91/42_works/minishell	\$>	0	
	\$> export CDPATH=	NON GERE		OK	OK	\$>	0		A gerer ?
	\$> cd \$HOME/..	NON GERE		OK	OK	/home	\$>	0	/home
	\$> export CDPATH=	NON GERE		OK	OK	/home/vietdu91	\$>	0	A gerer ?
	\$> cd /home/vietdu91	NON GERE		OK	OK	\$>	0		/home/vietdu91
	\$> export CDPATH=.	NON GERE		OK	OK	\$>	0		A gerer ?
	\$> cd	NON GERE		OK	OK	\$>	0		/home/vietdu91/42_works/minishell
	\$> export CDPATH=.	NON GERE		OK	OK	\$>	0		A gerer ?
	\$> cd ..	NON GERE		OK	OK	/home/vietdu91/42_works	\$>	0	/home/vietdu91/42_works
BATAARDS	\$> chmod 000 minishell			OK	OK	bash: ./minishell : Permission denied	\$>	126	
	\$> ./minishell			OK	OK	\$>	0		
	\$> 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	NON GERE		OK	OK	\$>	0		
	\$> ./minishell			OK	OK	\$>	0		
	\$> env grep SHLVL			OK	OK	\$>	0		
	\$> exit			OK	OK	\$>	0		
	\$> env grep SHLVL			OK	OK	\$>	0		
	\$> touch hola			OK	OK	bash: ./hola: Permission denied	\$>	126	
	\$> ./hola			OK	OK	\$>	66		
	\$> env wc -l			OK	OK	\$>	0		
	\$> env wc "-l			OK	OK	bash: wc -l: command not found	\$>	127	
	\$> expr 1 + 1			OK	OK	\$>	0		
	\$> expr \$? + \$?			OK	OK	0	\$>	0	Fait la somme de la valeur de \$?
	\$> env -i ./minishell			OK	OK	PWD=/mnt/nfs/homes/entran/Desktop/42_minishell	\$>	0	
	\$> env			OK	OK	SHLVL=1	\$>	0	
	\$> env -i ./minishell			OK	OK	/usr/bin/env	\$>	0	
	\$> export			OK	OK	export QLODPWD	\$>	0	
	\$> export			OK	OK	export PWD="/mnt/nfs/homes/entran/Desktop/42_minishell"	\$>	0	
	\$> export			OK	OK	export SHLVL="1"	\$>	0	
	\$> env -i ./minishell			OK	OK	bash: cd: HOME not set	\$>	1	
	\$> cd			OK	OK	\$>	0		/home/vietdu91
	\$> env -i ./minishell	NON GERE		OK	OK	\$>	0		
	\$> cd -			OK	OK	\$>	0		
	\$> exit			OK	OK	exit	~/minishell \$>	0	
	\$> exit exit			OK	OK	exit	bash: exit: exit: numeric argument required	2	
	\$> exit hola			OK	OK	~/minishell \$>	2		
	\$> exit hola			OK	OK	exit	bash: exit: hola: numeric argument required	2	
	\$> exit hola que tal			OK	OK	~/minishell \$>	2		
	\$> exit 42			OK	OK	exit	bash: exit: 42: numeric argument required	42	
	\$> exit 000042			OK	OK	~/minishell \$>	42		
	\$> exit 666			OK	OK	exit	~/minishell \$>	154	
	\$> exit 666 666			OK	OK	exit	~/minishell \$>	1	
	\$> exit 666 666 666			OK	OK	bash: exit: too many arguments	\$>	1	
	\$> exit -666 666			OK	OK	exit	~/minishell \$>	1	
	\$> exit hola 666			OK	OK	bash: exit: too many arguments	\$>	1	
	\$> exit 666 666 666 666			OK	OK	exit	~/minishell \$>	2	
	\$> exit 666 666 666 666 666			OK	OK	bash: exit: hola: numeric argument required	~/minishell \$>	1	
	\$> exit 666 666 666 666 666 666			OK	OK	exit	~/minishell \$>	1	
	\$> exit 666 666 666 666 666 666 666			OK	OK	bash: exit: too many arguments	\$>	1	

\$> exit 666 hola 666			OK	OK	exit	1		
\$> exit hola 666 666			OK	OK	bash: exit: too many arguments	2		
\$> exit 259			OK	OK	exit	3		
\$> exit -4			OK	OK	bash: exit: hola: numeric argument required	252		
\$> exit -42			OK	OK	~/minishell \$>	214		
\$> exit -0000042			OK	OK	exit	214		
\$> exit -259			OK	OK	~/minishell \$>	253		
\$> exit -666			OK	OK	exit	102		
\$> exit +666			OK	OK	~/minishell \$>	154		
\$> exit 0			OK	OK	exit	0		
\$> exit +0			OK	OK	~/minishell \$>	0		
\$> exit -0			OK	OK	exit	0		
\$> exit +42			OK	OK	~/minishell \$>	42		
\$> exit -69 -96			OK	OK	exit	1		
\$> exit --666			OK	OK	bash: exit: too many arguments	2		
\$> exit +++666			OK	OK	exit	2		
\$> exit ++++0			OK	OK	bash: exit: --666: numeric argument required	2		
\$> exit -----0			OK	OK	~/minishell \$>	2		
\$> exit "666"			OK	OK	exit	154		
\$> exit '666'			OK	OK	~/minishell \$>	154		
\$> exit '-666'			OK	OK	exit	102		Marche aussi pour les double quotes
\$> exit '+666'			OK	OK	~/minishell \$>	154		Marche aussi pour les double quotes
\$> exit '...666'			OK	OK	exit	2		Marche aussi pour les double quotes
\$> exit '+++666'			OK	OK	bash: exit: ---666: numeric argument required	2		Marche aussi pour les double quotes
\$> exit '6'66			OK	OK	~/minishell \$>	154		Marche aussi pour les double quotes
\$> exit '2'66'32'			OK	OK	exit	8		Marche aussi pour les double quotes
\$> exit "'666'"			OK	OK	~/minishell \$>	2		
\$> exit '"666"'			OK	OK	exit	2		
\$> exit '666'"666'666			OK	OK	bash: exit: "666": numeric argument required	170		
\$> exit +'666'"666'666			OK	OK	~/minishell \$>	170		
\$> exit -'666'"666'666			OK	OK	exit	86		
\$> exit 9223372036854775807			OK	OK	~/minishell \$>	255		
\$> exit 9223372036854775808			OK	OK	exit	2		
\$> exit -9223372036854775808			OK	OK	bash: exit: 9223372036854775808: numeric argument required	0		
\$> exit -9223372036854775809			OK	OK	~/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
\$> ls exit 42			OK	OK	\$>	42		Chaque commande d'un pipeline est exécutée dans son propre sous-shell, qui est un processus distinct
\$> exit ls			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c	0		Ne fait rien
\$> echo hola > bonjour			OK	OK	hola\$	0		
\$> exit cat -e bonjour			OK	OK	\$>	0		
\$> echo hola > bonjour			OK	OK	\$>	0		
\$> 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	\$>	0		
\$> echo out cat -e			OK	OK	out\$	0		
\$> echo out echo non echo hola grep out			OK	OK	\$>	0		
\$> echo out echo non echo hola grep non			OK	OK	\$>	0		
\$> echo out 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=4899<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			OK	OK	vietdu91	0		
\$> cat /tmp/bonjour			OK	OK	\$>	0		
\$> whoami cat -e cat -e > /tmp/bonjour			OK	OK	vietdu91\$	0		
\$> cat /tmp/bonjour			OK	OK	\$>	0		
\$> whereis ls cat -e cat -e > /tmp/bonjour			OK	OK	ls: /usr/bin/ls /usr/share/man/man1/ls.1posix.gz /usr/share/man/man1/ls.1.gz\$	0		
\$> cat /tmp/bonjour			OK	OK	\$>	127		
\$> ls hola			OK	OK	bash: hola: command not found	2		
\$> ls ls hola			OK	OK	\$>	127		
\$> ls ls hola			OK	OK	bash: hola: command not found	0		
\$> ls hola ls			OK	OK	bash: hola: command not found	0		
\$> ls ls hola rev			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c	0		
\$> ls ls echo hola rev			OK	OK	bash: hola: command not found	0		
\$> cat cat cat ls			OK	OK	\$>	0		

[illegible]

\$> 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]	
\$> (ls pwd)&&(ls 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]	
\$> (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 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' 1 1 34 \$>	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' 1 1 34 \$>	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			
\$> echo que tal >> bonjour \$> cat bonjour			OK	OK	hola que tal \$>	0 0			

\$> echo hola > bonjour					\$>	0		
\$> echo que tal >> bonjour					\$>	0		
\$> cat < bonjour			OK	OK	\$> hola que tal	0		
\$> echo hola > bonjour					\$>	0		
\$> rm bonjour					que tal	0		
\$> echo que tal >> bonjour			OK	OK	\$>	0		
\$> cat < bonjour					\$>	0		
\$> echo hola que tal > bonjour			OK	OK	hola que tal	0		
\$> cat bonjour					\$>	0		
\$> echo hola que tal > /tmp/bonjour			OK	OK	hola que tal\$	0		
\$> cat -e /tmp/bonjour					\$>	0		
\$> export HOIA=hey			OK	OK	\$>	0		Ne fait rien
\$> echo bonjour > \$HOIA					\$> hola	0		
\$> echo \$HOIA					\$>	0		
\$> whereis grep > Docs/bonjour			OK	OK	grep: /usr/bin/grep /usr/share/man/man1/grep.1.gz	0		
\$> cat Docs/bonjour					/usr/share/man/man1/grep.1postx.gz /usr/share/info/grep.info.gz	0		
\$> ls -la > Docs/bonjour					\$>	0		
\$> cat Docs/bonjour			OK	OK	total 16 drwx----- 2 vietdu91 [...] . drwxrwxrwt 51 root [...] .. -rwxr-xr-x 8 vietdu91 [...] crashtest.c [...]	0		
\$> pwd-bonjour					\$>	0		
\$> cat bonjour			OK	OK	/home/vietdu91/42_works/minishell	0		
\$> pwd > bonjour			OK	OK	/home/vietdu91/42_works/minishell	0		
\$> cat bonjour					\$>	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			OK	OK	hola	0		
\$> cat bonjour					\$>	0		
\$> > bonjour echo hola			OK	OK	hola	0		
\$> cat bonjour					\$>	0		
\$> prout hola > bonjour			OK	OK	bash: prout: command not found	127		Erreur mais file "bonjour" cree tout de meme
\$> ls					\$> bonjour [...]	0		
\$> echo hola > hello >> hello >> hello					\$>	0		
\$> ls			OK	OK	\$> hello	0		
\$> cat hello					\$> hola	0		
\$> echo hola > hello >> hello >> hello			OK	OK	\$>	0		
\$> echo hola >> hello					hola	0		
\$> echo hola > hello >> hello >> hello			OK	OK	\$>	0		
\$> cat < hello					hola	0		
\$> echo hola >> hello >> hello > hello			OK	OK	hola	0		
\$> echo hola >> hello					\$>	0		
\$> cat < hello					hola	0		
\$> echo hola >> hello >> hello > hello			OK	OK	hola	0		
\$> echo hola >> hello					\$>	0		
\$> cat < hello					hola	0		
\$> echo hola > hello			OK	OK	hola	0		
\$> echo hola >> hello >> hello >> hello					hola	0		
\$> echo hola >> hello					hola	0		
\$> cat < hello					\$>	0		
\$> echo hola > hello			OK	OK	\$>	0		
\$> echo hey > bonjour					\$>	0		
\$> echo <bonjour <hello					\$>	0		
\$> echo hola > hello			OK	OK	\$>	0		
\$> echo hey > bonjour					\$>	0		
\$> echo <hello <bonjour					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> rm bonjour hello					\$>	0		
\$> echo hola > bonjour > hello > bonjour			OK	OK	\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> echo hola > bonjour > hello > bonjour					\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> rm bonjour hello					\$>	0		
\$> echo hola > bonjour > hello >> bonjour			OK	OK	\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> rm bonjour hello					\$>	0		
\$> echo hola >> bonjour >> hello >> bonjour			OK	OK	\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> rm bonjour hello					\$>	0		
\$> echo hola >> bonjour >> hello >> bonjour			OK	OK	\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> echo hola > bonjour			OK	OK	hola	0		
\$> echo hey > hello					\$>	0		
\$> rm bonjour hello					\$>	0		
\$> echo hola >> bonjour >> hello >> bonjour			OK	OK	\$>	0		
\$> cat bonjour					\$>	0		
\$> cat hello					\$>	0		
\$> > bonjour echo hola bonjour			OK	OK	bonjour hola	0		
\$> cat bonjour					\$>	0		
\$> >bonjour echo > hola>bonjour>hola>>bonjour>hola hey >bonjour			OK	OK	\$>	0		
\$> hola >hola					\$>	0		
\$> cat bonjour					hey hola	0		
\$> cat hola					\$>	0		
\$> echo bonjour > hola1			OK	OK	hola1 hola2 [...]	0		
\$> echo hello > hola2					\$>	0		
\$> echo 2 >hola1 >> hola2					\$>	0		
\$> ls					hello	0		
\$> cat hola1					2	0		
\$> cat hola2					\$>	0		
\$> echo bonjour > hola1			OK	OK	hola1 hola2 [...]	0		
\$> echo hello > hola2					\$>	0		
\$> echo 2 >>hola1 > hola2					bonjour	0		
\$> ls					\$>	0		
\$> cat hola1					2	0		
\$> cat hola2					\$>	0		
\$> > pwd			OK	OK	pwd	0		
\$> ls					\$>	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	\$>	1		
\$> cat <srcs/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	\$>	1		
\$> cat >>			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				bash: ls2: No such file or directory \$>	1		
» ls		OK	OK	ls1 [...] \$>	0		
» >>hola		OK	OK	\$>	0		
» cat hola				\$>	0		
» echo hola > bonjour		OK	OK	hola \$>	0		
» cat < bonjour				\$>	0		
» echo hola >bonjour		OK	OK	hola \$>	0		
» cat <bonjour				\$>	0		
» echo hola>bonjour		OK	OK	hola \$>	0		
» cat<bonjour				\$>	0		
» echo hola > >bonjour		OK	OK	hola \$>	0		
» cat< > > bonjour		OK	OK	hola \$>	0		
» echo hola > srcs/bonjour		OK	OK	hola \$>	0		
» cat < srcs/bonjour				\$>	0		
» echo hola >srcs/bonjour		OK	OK	hola \$>	0		
» cat <srcs/bonjour				\$>	0		
» echo hola > bonjour				hola \$>	0		
» echo que tal >> bonjour		OK	OK	que tal \$>	0		
» cat < bonjour				\$>	0		
» echo hola > bonjour				que tal \$>	0		
» rm bonjour		OK	OK	\$>	0		
» echo que tal >> bonjour				\$>	0		
» cat < bonjour				\$>	0		
» e'c'h'o hola > bonjour		OK	OK	hola \$>	0		
» cat 'bo'n'jour				\$>	0		
» echo hola > bonjour\ 1	NON GERE	OK	OK	bonjour 1' [...] \$>	0		
» ls				hola \$>	0		
» cat bonjour\ 1				\$>	0		
» echo hola > bonjour hey				bonjour [...] \$>	0		
» ls				\$>	0		
» cat bonjour		OK	OK	hola hey \$>	0		
» cat hey				cat: hey: No such file or directory \$>	1		
» echo hola > srcs/bonjour				\$>	0		
» echo hey > srcs/hello		OK	OK	bash: prout: No such file or directory \$>	0		
» srcs/bonjour srcs/hello <prout				\$>	1		
» cat srcs/bonjour srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	0		
» rm srcs/bonjour srcs/hello		OK	OK	bonjour hello \$>	0		
» srcs/bonjour srcs/hello <prout				\$>	0		
» ls srcs				\$>	0		
» cat srcs/bonjour srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour <prout >srcs/hello		OK	OK	hey \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» rm srcs/bonjour srcs/hello		OK	OK	bonjour \$>	0		
» srcs/bonjour <prout >srcs/hello				\$>	1		
» ls ..				\$>	0		
» cat ../bonjour ../hello				\$>	0		
» echo hola > ../bonjour		OK	OK	bash: prout: No such file or directory \$>	0		
» echo hey > ../hello				\$>	0		
» >../bonjour >../hello <prout				\$>	1		
» cat ../bonjour ../hello				\$>	0		
» echo hola > ../bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > ../hello				\$>	0		
» rm ../bonjour ../hello		OK	OK	bonjour \$>	0		
» >../bonjour <prout >../hello				\$>	1		
» ls ..				\$>	0		
» cat ../bonjour				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour >>srcs/hello <prout		OK	OK	hey \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour <prout >srcs/hello		OK	OK	bonjour \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour <prout >>srcs/hello		OK	OK	hey \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour <prout >>srcs/hello		OK	OK	bonjour \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > srcs/bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > srcs/hello				\$>	1		
» srcs/bonjour <prout >>srcs/hello		OK	OK	hey \$>	0		
» cat srcs/bonjour				\$>	0		
» cat srcs/hello				\$>	0		
» echo hola > bonjour				hola \$>	0		
» echo hey > hello		OK	OK	>> \$>	0		
» <bonjour >hello				\$>	0		
» cat bonjour				\$>	0		
» cat hello				\$>	0		
» echo hola > bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > hello		OK	OK	\$>	1		
» >bonjour >hello < prout				\$>	0		
» cat bonjour				\$>	0		
» cat hello				\$>	0		
» echo hola > bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > hello				\$>	0		
» rm bonjour hello		OK	OK	\$>	1		
» <bonjour >hello < prout				\$>	0		
» cat bonjour				\$>	1		
» cat hello				\$>	0		
» echo hola > bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > hello				\$>	0		
» >bonjour <prout hello		OK	OK	hey \$>	1		
» cat bonjour				\$>	0		
» cat hello				\$>	0		
» echo hola > bonjour				bash: prout: No such file or directory \$>	0		
» echo hey > hello				\$>	0		
» rm bonjour hello		OK	OK	\$>	1		
» <bonjour <prout hello				\$>	0		
» cat bonjour				\$>	1		
» cat hello				\$>	0		
» echo hola > bonjour				bash: prout: No such file or directory \$>	0		
» <bonjour cat wc > bonjour		OK	OK	\$>	0		
» cat bonjour				\$>	0		
» cat hello				\$>	0		

WILDI	\$> ls *.hola	NON GERE		OK	OK	ls: cannot access '*.hola' : No such file or directory \$>	2	[BONUS]	
	\$> cat M*le	NON GERE		OK	OK	[offfiche le Makefile] \$>	0	[BONUS]	
	\$> cat M*ee	NON GERE		OK	OK	cat: 'M*ee': No such file or directory \$>	1	[BONUS]	
	\$> cat Make*file	NON GERE		OK	OK	[offfiche le Makefile] \$>	0	[BONUS]	