

# MINISHELL

**[FR] ATTENTION**

**Les comportements présents dans cet Excel ne sont pas tous à gérer dans la Mandatory du projet Minishell !**

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

**Vérifiez 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 êtes chauds pour tout gérer, GG à vous !)**

**Bon courage et n'hésitez pas à m'envoyer un MP si besoin !**

**[EN] WARNING**

**The behaviors listed in this Excel file are not all required to be handled in the mandatory part of the Minishell project!**

**Don't stress yourself out trying to handle all of them!**

**Just make sure that these unmanageable cases don't cause a segfault or exhibit suspicious behavior!**

Use your common sense to make the best use of this document and create a Minishell that is not overly ambitious, depending on your project situation.

**(However, if you're up for handling everything, GG to you!)**

**Good luck, and feel free to DM me if needed!**

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

---\*/\*\*/

		A faire	A tester	Leaks ?		Comportement attendu	Sortie d'exit (echo \$?)	Notes
		X	X	OK KO	OK KO			
\$> \n (touche entrée)					ok	\$>	Exit de l'ancienne commande	
\$> [que des espaces]					ok		0	
\$> [que des tabulations]					ok		0	Crl + V + Tab
\$> :					ok		0	Marche aussi pour "#"
\$> !					KO		1	
\$> >					ok	bash: syntax error near unexpected token `newline` \$>	2	
\$> <					ok		2	
\$> >>					ok		2	
\$> <<					ok		2	
\$> <>					KO		2	
\$> >>>>					ok	bash: syntax error near unexpected token `>>` \$>	2	
\$> >>>>>>>>>>>>					ok	bash: syntax error near unexpected token `>>` \$>	2	
\$> <<<<<					ok	bash: syntax error near unexpected token `<<` \$>	2	



\$> ( ( ( ( ) ) ) )	NON GERE					bash: syntax error near unexpected token `)' \$>	2	
\$> ""	NON GERE					bash: : command not found \$>	127	
\$> "hola"					ok	bash: hola: command not found \$>	127	
\$> 'hola'					ok	bash: hola: command not found \$>	127	
\$> ''	NON GERE					bash: : command not found \$>	127	
\$> *	NON GERE					bash: <b>crashtest.c</b> : command not found \$>	127	[BONUS]
\$> */*	NON GERE					bash: <b>Docs/attest.cation</b> : command not found \$>	127	[BONUS]
\$> */*	NON GERE					bash: */*: No such file or directory \$>	127	[BONUS] S'il n'y aucun dossier à l'intérieur
\$> .	NON GERE					bash: .: filename argument required \$>	2	
\$> ..	NON GERE					bash: ..: command not found \$>	127	
\$> ~	NON GERE					bash: <b>/home/vietdu91</b> : Is a directory \$>	126	
\$> ABC=hola	NON GERE					\$>	0	/!\ A gerer comme une erreur
\$> 4ABC=hola	NON GERE					bash: 4ABC=hola: command not found \$>	127	
hola					ok	bash: hola: command not found \$>	127	
\$> hola que tal					ok	bash: hola: command not found \$>	127	
\$> Makefile					ok	bash: Makefile: command not found \$>	127	
\$> echo					ok	\$>	0	
\$> echo -n					ok	\$>	0	
\$> echo Hola					ok	Hola \$>	0	
\$> echoHola					ok	bash: echoHola: command not found \$>	127	
\$> echo-nHola					OK	bash: echo-nHola: command not found \$>	127	
\$> echo -n Hola					ok	Hola\$>	0	
\$> echo "-n" Hola					OK	Hola\$>	0	
\$> echo -nHola					ok	-nHola \$>	0	
\$> echo Hola -n					ok	Hola -n \$>	0	
\$> echo Hola Que Tal					ok	Hola Que Tal \$>	0	
\$> echo        Hola					ok	Hola \$>	0	
\$> echo    Hola        Que        Tal					ok	Hola Que Tal \$>	0	
\$> echo        \n hola					KO	n hola \$>	0	A gerer ?
\$> echo "        "   cat -e					ok	\$ \$>	0	
\$> echo          cat -e					ok	\$ \$>	0	
\$> ""'echo hola'""' que'""' tal'""'					OK	hola que tal \$>	0	
\$> echo -n -n					ok	\$>	0	
\$> echo -n -n Hola Que					ok	Hola Que\$>	0	
\$> echo -p					ok	-p \$>	0	
\$> echo -nnnnn					ok	\$>	0	
\$> echo -n -nnn -nnnn					ok	\$>	0	

\$> echo -n-nnn -nnnn					ok	-n-nnn -nnnn \$>	0		
\$> echo -n -nnn hola -nnnn					ok	hola -nnnn\$>	0		
\$> echo -n -nnn-nnnn					ok	-nnn-nnnn\$>	0		
\$> echo -----n					ok	-----n \$>	0		
\$> echo -nnn -----n					ok	-----n\$>	0		
\$> echo -nnn -----nn---nnnn					ok	-----nn---nnnn\$>	0		
\$> echo -nnn -----nnnn					ok	-----nnnn\$>	0		
\$> echo \$					ok	\$ \$>	0		
\$> echo \$?					ok	0 \$>	0		Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
\$> echo \$?\$					ok	0\$ \$>	0		
\$> echo \$?   echo \$?   echo \$?					ok	0 \$>	0		
\$> echo \$:\$=   cat -e					ok	:\$:\$= \$>	0		
\$> echo " \$ "   cat -e					ok	\$ \$ \$>	0		
\$> echo ' \$ '   cat -e					ok	\$ \$ \$>	0		
\$> echo \$HOME					ok	/home/vietdu91 \$>	0		
\$> echo \ \$HOME					ok	\$HOME \$>	0		
\$> echo my shit terminal is [\$TERM]					ok	my shit terminal is [xterm-256color] \$>	0		
\$> echo my shit terminal is [\$TERM4					ok	my shit terminal is [ \$>	0		
\$> echo my shit terminal is [\$TERM4]					ok	my shit terminal is [] \$>	0		
\$> echo \$UID					ok	1000 \$>	0		
\$> echo \$HOME9					ok	 \$>	0		Marche pour tous les caractères alphanumériques de l'ASCII
\$> echo \$9HOME					ok	HOME \$>	0		
\$> echo \$HOME%					ok	/home/vietdu91% \$>	0		Marche pour les autres caractères (ponctuation, caractères non-ASCII, etc.)
\$> echo \$UID\$HOME					ok	1000/home/vietdu91% \$>	0		
\$> echo Le path de mon HOME est \$HOME					ok	Le path de mon HOME est /home/vietdu91 \$>	0		
\$> echo \$USER\$var\$USER\$USER\ \$USERtest\$USER					ok	vietdu91\$USERvietdu91\$USERtestvietdu91 \$>	0		A ne pas gerer si '\\' non gere : Doit sortir : vietdu91vietdu91vietdu91vietdu91
\$> echo \$hola*					ok	 \$>	0		Cas d'un variable inexistant
\$> echo -nnnn \$hola					ok	\$>	0		
\$> echo > <					ok	bash: syntax error near unexpected token `<' \$>	2		
\$> echo					ok	bash: syntax error near unexpected token ` ' \$>	2		
\$> EechoE					ok	bash: EechoE: command not found \$>	127		
\$> .echo.					ok	bash: .echo.: command not found \$>	127		
\$> >echo>					ok	bash: syntax error near unexpected token `newline' \$>	2		
\$> <echo<					ok	bash: syntax error near unexpected token `newline' \$>	2		
\$> >>echo>>					ok	bash: syntax error near unexpected token `newline' \$>	2		
\$>  echo					ok	bash: syntax error near unexpected token ` ' \$>	2		
\$>  echo -n hola					ok	bash: syntax error near unexpected token ` ' \$>	2		

ECHO	\$> echo *	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo '*'	NON GERE				* \$>	0	Marche aussi pour les double quotes
	\$> echo D*	NON GERE				Docs Dockers Drawings \$>	0	[BONUS]
	\$> echo *Z	NON GERE				*Z \$>	0	[BONUS]
	\$> echo *t hola	NON GERE				crashtest.c test.c hola \$>	0	[BONUS]
	\$> echo *t	NON GERE				crashtest.c test.c \$>	0	[BONUS]
	\$> echo \$*	NON GERE				 \$>	0	[BONUS]
	\$> echo hola*hola *	NON GERE				hola*hola crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo \$hola*	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo \$HOME*	NON GERE				/home/vietdu91 \$>	0	[BONUS]
	\$> echo \$" "				ok	 \$>	0	
	\$> echo "\$" "				ok	\$ \$>	0	
	\$> echo '\$' '				ok	\$ \$>	0	
	\$> echo \$"HOME"				ok	HOME \$>	0	
	\$> echo \$' 'HOME				ok	HOME \$>	0	
	\$> echo \$" "HOME				ok	HOME \$>	0	
	\$> echo "\$HO"ME				ok	ME \$>	0	
	\$> echo '\$HO' ME				ok	\$HOME \$>	0	
	\$> echo "\$HO" "ME"				ok	ME \$>	0	
	\$> echo '\$HO' 'ME'				ok	\$HOME \$>	0	
	\$> echo "'\$HO' 'ME' "				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	/home/vietdu91 \$>	0	
	\$> echo "" \$HOME				ok	/home/vietdu91 \$>		
	\$> echo ' '\$HOME				ok	/home/vietdu91 \$>	0	
	\$> echo ' ' \$HOME				ok	/home/vietdu91 \$>		
	\$> echo \$"HO" "ME"				ok	HOME \$>	0	
	\$> echo \$'HO' 'ME'				ok	HOME \$>	0	
	\$> echo \$'HOME'				ok	HOME \$>	0	
	\$> echo "\$"HOME				ok	\$HOME \$>	0	
	\$> echo \$=HOME				ok	\$=HOME \$>	0	
	\$> echo \$"HOLA"				ok	HOLA \$>	0	
	\$> echo \$'HOLA'				ok	HOLA \$>	0	
	\$> echo \$DONTEXIST Hola				OK	Hola \$>	0	Le cas ou la variable n'existe pas et n'est pas importee
	\$> echo "hola"				ok	hola \$>	0	

\$> echo 'hola'					ok	hola \$>	0		
\$> echo ''hola''					ok	hola \$>	0		
\$> echo ''h'o'la''					ok	hola \$>	0		
\$> echo ''''h'o'la''''					ok	'h'o'la' \$>	0		
\$> echo ""'h'o'la'""					ok	hola' \$>	0		
\$> echo" 'hola' "					ok	bash: echo'hola': command not found \$>	0		
\$> echo ''hola''					ok	'hola' \$>	0		
\$> echo ""hola""					ok	"hola" \$>	0		
\$> echo ''ho''''l'a''''					ok	ho"l"a \$>	0		
\$> echo hola"''''''''''''''''					ok	hola \$>	0		
\$> echo hola"''''''''''''''''					ok	hola'''''''''' \$>	0		
\$> echo hola' ''''''''''''''''					ok	hola \$>	0		
\$> echo hola' ''''''''''''''''					ok	hola"'''''''''''' \$>	0		
\$> e"cho hola"					ok	bash: echo hola: command not found \$>	127		
\$> e'cho hola'					ok	bash: echo hola: command not found \$>	127		
\$> echo "hola     "   cat -e					ok	hola     \$ \$>	0		
\$> echo ""hola					ok	hola \$>	0		
\$> echo "" hola					ok	hola \$>	0		
\$> echo ""                hola					ok	hola \$>	0		
\$> echo ""hola					ok	hola \$>	0		
\$> echo "" hola					ok	hola \$>	0		
\$> echo hola""bonjour					ok	holabonjour \$>	0		
\$> "e"'c'ho 'b'"o"nj"o"'u'r					ok	bonjour \$>	0		
\$> ""e"'c'ho 'b'"o"nj"o"'u'r"					ok	bash: e'c'ho 'b'onjo'u'r: command not found \$>	127		
\$> echo "\$DONTEXIST"Makefile					ok	Makefile \$>	0		
\$> echo "\$DONTEXIST""Makefile"					ok	Makefile \$>	0		
\$> echo "\$DONTEXIST" "Makefile"					ok	Makefile \$>	0		Attention a l'espace avant
\$> \$?					ok	bash: 0: command not found \$>	127		Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
\$> \$?\$?					ok	bash: 00: command not found \$>	127		
\$> ?\$HOME					ok	bash: ?/hone/vietdu91: command not found \$>	127		
\$> \$					ok	bash: \$: command not found \$>	127		
\$> \$HOME					ok	bash: /home/vietdu91: Is a directory \$>	126		
\$> \$HOMEdskjhfkdshfsd					KO is directory	\$>	0		
\$> "\$HOMEdskjhfkdshfsd"					KO is directory	bash: : command not found \$>	127		
\$> '\$HOMEdskjhfkdshfsd'					ok	bash: \$HOMEdsdddfdfssfdfs: command not found \$>	127		
\$> \$DONTEXIST					KO is directory	\$>	0		Cas d'un variable inexistant

	\$> \$LESS\$VAR					ok	bash: -R: command not found \$>	127	
SIGNAUX	\$> Ctlr-C					OK	\$> ^C \$>	130	
	\$> holaCtlr-C					OK	\$> hola^C \$>	130	
	\$> cat (faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])					OK	^C \$>	130	
	\$> cat   rev(faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])					OK	^C \$>	0	
	\$> Ctlr-D					OK	\$> exit ~/minishell \$>	0	Sort du programme
	\$> hola Ctlr-D					OK	\$> hola	/	Ne fait rien
	\$> Ctlr-\					OK	\$>	/	Ne fait rien
	\$> hola Ctlr-\					OK	\$> hola	/	
	\$> cat (faire Ctlr-\ apres avoir fait plusieurs fois [ENTREE])					OK	^\\Quit (core dumped) \$>	131	
	\$> Ctlr-Z					OK	\$>	/	Ne fait rien
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-C une seconde apres)					OK	^C \$>	130	
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-D une seconde apres)					OK	\$>	0	Ne fait rien
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-\ une seconde apres)					OK	^\\Quit (core dumped) \$>	131	
	\$> env					OK	... PATH=... \$>	0	
	\$> env hola	NON GERE					env: 'hola': No such file or directory \$>	127	
	\$> env hola que tal	NON GERE					env: 'hola': No such file or directory \$>	127	
	\$> env env	NON GERE					... PATH=... \$>	0	Affiche une seule fois env
	\$> env env env env env	NON GERE					... PATH=... \$>	0	Affiche une seule fois env
	\$> env ls	NON GERE					crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	
	\$> env ./Makefile	NON GERE					env: './Makefile': Permission denied \$>	126	
	\$> export HOLA=bonjour \$> env					OK	... HOLA=bonjour \$>	0 0	
	\$> export HOLA=bonjour \$> env					OK	... HOLA=bonjour \$>	0 0	
	\$> export					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 \$> export					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	... export HOME="/home/vietdu91" export Hola9hey export PATH="..." \$>	0 0	

\$> export \$DONTEXIST				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	export HOME="/home/vietdu91" \$>	0	
\$> export ""				OK	bash: export: `': not a valid identifier \$>	1	
\$> export =				OK	bash: export: '=': not a valid identifier \$>	1	
\$> export %				OK	bash: export: '%': not a valid identifier \$>	1	
\$> export \$?				OK	bash: export: `0': not a valid identifier \$>	1	
\$> export ~=2				OK	bash: export: `?=2': not a valid identifier \$>	1	
\$> export 9HOLA=				OK	bash: export: `9HOLA=': not a valid identifier \$>	1	
\$> export HOLA9=bonjour \$> env				OK	... HOLA9=bonjour \$>	0 0	
\$> export _HOLA=bonjour \$> env				OK	... _HOLA=bonjour \$>	0 0	
\$> export ____HOLA=bonjour \$> env				OK	... ____HOLA=bonjour \$>	0 0	
\$> export _HO_LA_=bonjour \$> env				OK	... _HO_LA_=bonjour \$>	0 0	
\$> export HOL@=bonjour				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				bash: export: `HOL~A=bonjour': not a valid identifier \$>	1	
\$> export -HOLA=bonjour				OK	bash: export: -H: invalid option	2	
\$> export --HOLA=bonjour				OK	bash: export: -: invalid option	2	
\$> export HOLA-=bonjour				OK	bash: export: `HOLA-=bonjour': not a valid identifier \$>	1	
\$> export HO-LA=bonjour				OK	bash: export: `HO-LA=bonjour': not a valid identifier \$>	1	
\$> export HOL.A=bonjour				OK	bash: export: `HOL.A=bonjour': not a valid identifier \$>	1	
\$> export HOL\\\$A=bonjour	NON GERE				bash: export: `HOL\\\$A=bonjour': not a valid identifier \$>	1	
\$> export HO\\\$LA=bonjour	NON GERE				bash: export: `HO\\\$LA=bonjour': not a valid identifier \$>	1	
\$> export HOL}A=bonjour				OK	bash: export: `HOL}A=bonjour': not a valid identifier \$>	1	
\$> export HOL{A=bonjour				OK	bash: export: `HOL{A=bonjour': not a valid identifier \$>	1	
\$> export HO*LA=bonjour				OK	bash: export: `HO*LA=bonjour': not a valid identifier \$>	1	
\$> export HO#LA=bonjour				OK	bash: export: `HO#LA=bonjour': not a valid identifier \$>	1	
\$> export HO@LA=bonjour				OK	bash: export: `HO@LA=bonjour': not a valid identifier \$>	1	
\$> export HO!LA=bonjour				OK	bash: !LA=bonjour: event not found	0	
\$> export HO\$?LA=bonjour \$> env				OK	... HO0LA=bonjour \$>	0	
\$> export +HOLA=bonjour				OK	bash: export: `+HOLA=bonjour': not a valid identifier \$>	1	
\$> export HOL+A=bonjour				OK	bash: export: `HOL+A=bonjour': not a valid identifier \$>	1	
\$> export HOLA+=bonjour \$> env	NON GERE				... HOLA=bonjour \$>	0	
\$> export HOLA=bonjour \$> export HOLA+=bonjour \$> env	NON GERE				... HOLA=bonjourbonjour \$>	0	"+=" s'ajoute au contenu de la variable



ENV & EXPORT & UNSET	<pre>\$&gt; export HOLA=bonjour \$&gt; env</pre>	NON GERE					<pre>... PATH=... \$&gt;</pre>	0		
	<pre>\$&gt; export HOLA =bonjour</pre>					OK	<pre>bash: export: '=bonjour': not a valid identifier \$&gt;</pre>	1		
	<pre>\$&gt; export HOLA = bonjour</pre>					OK	<pre>bash: export: '=': not a valid identifier \$&gt;</pre>	1		
	<pre>\$&gt; export HOLA=bon jour \$&gt; env</pre>					OK	<pre>... HOLA=bon \$&gt;</pre>	0		
	<pre>\$&gt; export HOLA= bonjour \$&gt; env</pre>					OK	<pre>... HOLA= \$&gt;</pre>	0		
	<pre>\$&gt; export HOLA=bonsoir \$&gt; export HOLA=bonretour \$&gt; export HOLA=bonjour \$&gt; env</pre>					OK	<pre>... HOLA=bonjour \$&gt;</pre>	0 0 0		
	<pre>\$&gt; export HOLA=\$HOME \$&gt; env</pre>					OK	<pre>... HOLA=/home/vietdu91 \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bonjour\$HOME \$&gt; env</pre>					OK	<pre>... HOLA=bonjour/home/vietdu91 \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=\$HOMEbonjour \$&gt; env</pre>					OK	<pre>... HOLA= \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bon\$jour \$&gt; env</pre>					OK	<pre>... HOLA=bon \$&gt;</pre>	0 0		Le cas où la variable "\$jour" n'existe pas
	<pre>\$&gt; export HOLA=bon\jour \$&gt; env</pre>	NON GERE					<pre>... HOLA=bonjour \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bon\\jour \$&gt; env</pre>	NON GERE					<pre>... HOLA=bon\jour \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bon(jour</pre>	NON GERE					<pre>bash: syntax error near unexpected token `(' \$&gt;</pre>	2		Marche pour les caractères : ( )
	<pre>\$&gt; export HOLA=bon()jour</pre>	NON GERE					<pre>bash: syntax error near unexpected token `(' \$&gt;</pre>	2		
	<pre>\$&gt; export HOLA=bon&amp;jour</pre>	NON GERE					<pre>bash: jour: command not found \$&gt;</pre>	127		Marche pour les caractères : &
	<pre>\$&gt; export HOLA=bon@jour \$&gt; env</pre>					OK	<pre>\$&gt; env ... HOLA=bon@jour \$&gt;</pre>	0		
	<pre>\$&gt; export HOLA=bon;jour \$&gt; env</pre>					OK	<pre>bash: jour: command not found \$&gt; env ... HOLA=bon \$&gt;</pre>	127 0		
	<pre>\$&gt; export HOLA=bon!jour</pre>					OK	<pre>bash: !jour: event not found \$&gt;</pre>	1		
	<pre>\$&gt; export HOLA=bon"jour" \$&gt; env</pre>					OK	<pre>... HOLA=bonjour PATH=... \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA\$USER=bonjour \$&gt; env</pre>					OK	<pre>... HOLAvietdu91=bonjour PATH=... \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bonjour=casse-toi \$&gt; echo \$HOLA</pre>					OK	<pre>HOLA=bonjour=casse-toi \$&gt;</pre>	0 0		
	<pre>\$&gt; export "HOLA=bonjour"=casse-toi \$&gt; echo \$HOLA</pre>					OK	<pre>HOLA=bonjour=casse-toi \$&gt;</pre>	0 0		
	<pre>\$&gt; export HOLA=bonjour \$&gt; export BYE=casse-toi \$&gt; echo \$HOLA et \$BYE</pre>					OK	<pre>bonjour et casse-toi \$&gt;</pre>	0 0 0		
	<pre>\$&gt; export HOLA=bonjour BYE=casse-toi \$&gt; echo \$HOLA et \$BYE</pre>					OK	<pre>bonjour et casse-toi \$&gt;</pre>	0 0		
	<pre>\$&gt; export \$HOLA=bonjour \$&gt; env</pre>					OK	<pre>... bonjour=bonjour PATH=... \$&gt;</pre>	0		

<pre>\$&gt; export HOLA="bonjour  " \$&gt; echo \$HOLA   cat -e</pre>					OK	<pre>bonjour\$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="  -n bonjour  " \$&gt; echo \$HOLA</pre>					OK	<pre>bonjour\$&gt;</pre>	0		
<pre>\$&gt; export HOLA="bonjour  "/" \$&gt; echo \$HOLA</pre>					OK	<pre>bonjour / \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="'" \$&gt; echo " \$HOLA "   cat -e</pre>					OK	<pre>" \$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=at \$&gt; c\$HOLA Makefile</pre>					OK	<pre>[cela affiche le Makefile] \$&gt;</pre>	0		
<pre>\$&gt; export "" HOLA=bonjour \$&gt; env</pre>					OK	<pre>bash: export: `': not a valid identifier \$&gt; env ... HOLA=bonjour \$&gt;</pre>	1		
<pre>\$&gt; export HOLA="cat Makefile   grep NAME" \$&gt; echo \$HOLA</pre>					OK	<pre>cat Makefile   grep NAME \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=hey \$&gt; echo \$HOLA\$HOLA\$HOLA=hey\$HOLA</pre>					OK	<pre>heyheyhey=heyhey \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo \$HOLA   cat -e</pre>					OK	<pre>bonjour hey\$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo ""\$HOLA""   cat -e</pre>					OK	<pre> bonjour hey \$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo wesh"\$HOLA"   cat -e</pre>					OK	<pre>wesh bonjour hey \$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo wesh""\$HOLA.</pre>					OK	<pre>wesh bonjour hey . \$&gt;</pre>	0		/*! Bien supprimer les espaces en trop au debut et a la fin du string d'export
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo wesh\$""\$HOLA.</pre>					OK	<pre>weshHOLA. \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey  " \$&gt; echo wesh\$"\$HOLA HOLA".</pre>					OK	<pre>weshHOLA HOLA. \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=" hola et \$HOLA" \$&gt; echo \$HOLA</pre>					OK	<pre>hola et bonjour \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=' hola et \$HOLA' \$&gt; echo \$HOLA</pre>					OK	<pre>hola et \$HOLA \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=" hola et \$HOLA"\$HOLA \$&gt; echo \$HOLA</pre>					OK	<pre>hola et bonjourbonjour \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="ls      -l      - a" \$&gt; echo \$HOLA</pre>					OK	<pre>ls -l - a \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="s -la" \$&gt; l\$HOLA</pre>					OK	<pre>total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root   [...] .. -rwxr-xr-x  8 vietdu91 [...] crashtest.c [...] \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="s -la" \$&gt; l"\$HOLA"</pre>					OK	<pre>bash: ls -la: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA="s -la" \$&gt; l'\$HOLA'</pre>					OK	<pre>bash: l\$HOLA: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA="l" \$&gt; \$HOLAs</pre>					OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; export HOLA="l" \$&gt; "\$HOLA"s</pre>					OK	<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$&gt;</pre>	0		
<pre>\$&gt; export HOL=A=bonjour \$&gt; env</pre>					OK	<pre>... HOL=A=bonjour \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="l" \$&gt; '\$HOLA's</pre>					OK	<pre>bash: \$HOLAs: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOL=A="" \$&gt; env</pre>					OK	<pre>... HOL=A= \$&gt;</pre>	0		
<pre>\$&gt; export TE+S=T \$&gt; env</pre>					OK	<pre>bash: export: `TE+S=T': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export ""=""</pre>					OK	<pre>bash: export: `=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export '='=''</pre>					OK	<pre>bash: export: `=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export "=="=="</pre>					OK	<pre>bash: export: `===': not a valid identifier \$&gt;</pre>	1		

\$> export '='=''					OK	bash: export: `===': not a valid identifier \$>	1	
\$> export HOLA=p \$> export BYE=w \$> \$HOLA"BYE"d					OK	bash: pBYEd: command not found \$>	127	
\$> export HOLA=p \$> export BYE=w \$> "\$HOLA" '\$BYE' d					OK	bash: p\$BYEd: command not found \$>	127	
\$> export HOLA=p \$> export BYE=w \$> "\$HOLA""\$BYE"d					OK	/home/vietdu91/42_works/minishell	0	
\$> export HOLA=p \$> export BYE=w \$> \$"HOLA"\$"\$BYE"d					KO	bash: HOLABYEd: command not found \$>	127	
\$> export HOLA=p \$> export BYE=w \$> \$'HOLA'\$'\$BYE'd					KO	bash: HOLABYEd: command not found \$>	127	
\$> export HOLA=-n \$> "echo \$HOLA" hey					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	1'2\$C"\$D5"\$F'7'8 \$>	0 0	
\$> export HOLA=bonjour \$> env \$> unset HOLA \$> env						... PATH=... \$>	0 0 0 0	
\$> export HOLA=bonjour \$> env \$> unset HOLA \$> unset HOLA \$> env						... PATH=... \$>	0 0 0 0 0	
\$> unset PATH \$> echo \$PATH						\$>	0 0	
\$> unset PATH \$> ls						bash: ls: No such file or directory	127	
\$> unset ""						bash: unset: `': not a valid identifier \$>	1	
\$> unset INEXISTANT						\$>	0	Cas où la variable n'existe pas
\$> unset PWD \$> env   grep PWD \$> pwd						\$> \$> /home/vietdu91/42_works/minishell \$>	0 0 0	
\$> pwd \$> unset PWD \$> env   grep PWD \$> cd \$PWD \$> pwd						/home/vietdu91/42_works/minishell \$> \$> \$> \$> /home/vietdu91	0 0 0 0 0	
\$> unset OLDPWD \$> env   grep OLDPWD						\$> \$>	0 0	
\$> unset 9HOLA						bash: unset: `9HOLA': not a valid identifier \$>	1	
\$> unset HOLA9						\$>	0	
\$> unset HOL?A						bash: unset: `HOL?A': not a valid identifier \$>	1	
\$> unset HOLA HOL?A						bash: unset: `HOL?A': not a valid identifier \$>	1	Il unset bien HOLA
\$> unset HOL?A HOLA						bash: unset: `HOL?A': not a valid identifier \$>	1	Il unset bien HOLA
\$> unset HOL?A HOL.A						bash: unset: `HOL?A': not a valid identifier bash: unset: `HOL.A': not a valid identifier \$>	1	
\$> unset HOLA=						bash: unset: `HOLA=': not a valid identifier \$>	1	
\$> unset HOL\\A	NON GERE					bash: unset: `HOL\\A': not a valid identifier	1	
\$> unset HOL;A	NON GERE					bash: A: command not found	127	
\$> unset HOL.A						bash: unset: `HOL.A': not a valid identifier	1	
\$> unset HOL+A						bash: unset: `HOL+A': not a valid identifier	1	
\$> unset HOL=A						bash: unset: `HOL=A': not a valid identifier	1	
\$> unset HOL{A						bash: unset: `HOL{A': not a valid identifier	1	
\$> unset HOL}A						bash: unset: `HOL}A': not a valid identifier	1	
\$> unset HOL-A						bash: unset: `HOL-A': not a valid identifier	1	

	\$> unset -HOLA						bash: unset: -H: invalid option	2		
	\$> unset _HOLA						\$>	0		
	\$> unset HOL_A						\$>	0		
	\$> unset HOLA_						\$>	0		
	\$> unset HOL*A						bash: unset: `HOL*A': not a valid identifier	1		
	\$> unset HOL#A						bash: unset: `HOL#A': not a valid identifier	1		
	\$> unset \$HOLA						\$>	0		Il va unset le contenu de \$HOLA, c'est-a-dire la variable bonjour
	\$> unset \$PWD						bash: unset: `/home/vietdu91/42_works/minishell': not a valid identifier	1		
	\$> unset HOL@						bash: unset: `HOL@': not a valid identifier	1		
	\$> unset HOL!A						bash: !A: event not found	1		
	\$> unset HOL^A						bash: unset: `HOL^A': not a valid identifier	1		
	\$> unset HOL\$?A						\$>	0		Unset prend en compte le contenu de \$?, c'est-a-dire qu'il lit HOL0A ou HOL127A
	\$> unset HOL~\A						bash: unset: `HOL~': not a valid identifier	1		
	\$> unset "" HOLA						bash: unset: `': not a valid identifier	1		HOLA sera quand meme unset !!
	\$> env   grep HOLA						\$>	0		
	\$> unset PATH						\$>	0		
	\$> echo \$PATH						\$>	0		
	\$> unset PATH						bash: cat: No such file or directory	0		
	\$> cat Makefile						\$>	1		
	\$> unset =						bash: unset: `=': not a valid identifier	0		
	\$> unset =====						bash: unset: `=====': not a valid identifier	0		
	\$> unset ++++++						bash: unset: `++++++': not a valid identifier	0		
	\$> unset _____						\$>	0		
	\$> unset export						\$>	0		
	\$> unset echo						\$>	0		
	\$> unset pwd						\$>	0		
	\$> unset cd						\$>	0		
	\$> unset unset						\$>	0		
	\$> unset sudo						\$>	0		
	\$> export hola   unset hola   echo \$?						0	0		
FICHIERS BINAIRE	\$> /bin/echo					ok	\$>	0		
	\$> /bin/echo Hola Que Tal					ok	Hola Que Tal	0		
	\$> /bin/env					ok	... PATH=...	0		
	\$> /bin/cd Desktop					ok	bash: /bin/cd: No such file or directory	127		
HISTORIQUE	\$> history	NON GERE					... 558 echo \$? 559 clear 560 history	0		Affiche les 500 dernières commandes
	\$> [touche du haut]					ok	\$> clear	/		Defile l'historique
	\$> pwd					ok	/home/vietdu91/42_works/minishell	0		Retourne \$PWD
	\$> pwd hola					ok	/home/vietdu91/42_works/minishell	0		
	\$> pwd ./hola					ok	/home/vietdu91/42_works/minishell	0		
	\$> pwd hola que tal					ok	/home/vietdu91/42_works/minishell	0		
	\$> pwd -p					ok	bash: pwd: -p: invalid option	2		
	\$> pwd --p					ok	bash: pwd: -: invalid option	2		
	\$> pwd ---p					ok	bash: pwd: -: invalid option	2		
	\$> pwd -- p	NON GERE				KO	/home/vietdu91/42_works/minishell	0		

CD	PWD	\$> pwd pwd pwd				ok	/home/vietdu91/42_works/minishell	0		N'affiche qu'une fois
		\$> pwd ls				ok	/home/vietdu91/42_works/minishell	0		
		\$> pwd ls env				ok	/home/vietdu91/42_works/minishell	0		
		\$> cd				ok	\$>	0		On doit se retrouver dans le dossier défini par \$HOME
		\$> cd .				ok	\$>	0		L'user reste dans le même dossier
		\$> cd ./				ok	\$>	0		/home/vietdu91/42_works/minishell
		\$> cd ../../.				ok	\$>	0		/home/vietdu91/42_works/minishell
		\$> cd ../.././				ok	\$>	0		/home/vietdu91/42_works/minishell
		\$> cd ..				ok	\$>	0		On se retrouve au dossier précédent
		\$> cd ../				ok	\$>	0		/home/vietdu91/42_works
		\$> cd ../../				ok	\$>	0		/home/vietdu91
		\$> cd ../.				ok	\$>	0		/home/vietdu91/42_works
		\$> cd ../.././.				ok	\$>	0		/home/vietdu91/42_works
		\$> cd srcs				ok	\$>	0		/home/vietdu91/42_works/minishell/srcs
		\$> cd srcs objs				ok	bash: cd: too many arguments	1		/home/vietdu91/42_works/minishell
		\$> cd 'srcs'				ok	\$>	0		/home/vietdu91/42_works/minishell/srcs
		\$> cd "srcs"				ok	\$>	0		/home/vietdu91/42_works/minishell/srcs
		\$> cd '/etc'				ok	\$>	0		Marche aussi avec les double quotes
		\$> cd /e'tc'				ok	\$>	0		/etc
		\$> cd /e"tc"				ok	\$>	0		/etc
		\$> cd sr				ok	bash: cd: sr: No such file or directory	1		
		\$> cd Makefile				ok	bash: cd: Makefile: Not a directory	1		
		\$> cd ../minishell				ok	\$>	0		/home/vietdu91/42_works/minishell
		\$> cd ../../../../../../../..				ok	\$>	0		/
		\$> cd ../../../../../../../bin/ls				ok	bash: cd: ../../../../../../../bin/ls: No such file or directory	1		/home/vietdu91/42_works/minishell
		\$> cd /				ok	\$>	0		/
		\$> cd '/'				ok	\$>	0		Marche aussi avec les double quotes
		\$> cd //	NON GERE				//	0		/
		\$> pwd	NON GERE				\$>	0		/
		\$> cd '///'	NON GERE				//	0		Marche aussi avec les double quotes
		\$> pwd					\$>	0		/
		\$> cd ///				ok	/	0		/
		\$> pwd					\$>	0		/
		\$> cd //////////				ok	/	0		/
		\$> pwd					\$>	0		/
		\$> cd '/////////'				ok	/	0		Marche aussi avec les double quotes
		\$> pwd					\$>	0		/
		\$> cd /minishell				ok	bash: cd: /minishell: No such file or directory	1		
		\$> cd /				ok	\$>	0		/
		\$> cd ..				ok	\$>	0		
		\$> cd _				ok	bash: cd: _: No such file or directory	1		
		\$> cd -				ok	/home/vietdu91/42_works/minishell	0		Retourne à \$OLD_PWD \$OLD_PWD et \$PWD sont inversés
		\$> cd --	NON GERE			KO	\$>	0		/home/vietdu91/42_works/minishell
		\$> cd ---				ok	bash: cd: ---: invalid option	2		Retourne à \$HOME \$OLD_PWD et \$PWD sont inversés

\$> cd \$HOME					ok	\$>	0	/home/vietdu91
\$> cd \$HOME \$HOME					ok	bash: cd: too many arguments \$>	1	
\$> cd \$HOME/42_works					ok	\$>	0	/home/vietdu91/42_works
\$> cd "\$PWD/srcs"					ok	\$>	0	/home/vietdu91/42_works/minishell/srcs
\$> cd '\$PWD/srcs'					ok	bash: cd: \$PWD/srcs: No such file or directory \$>	1	
\$> unset HOME \$> cd \$HOME					ok	bash: cd: HOME not set \$>	1	
\$> unset HOME \$> export HOME= \$> cd					ok	\$>	0	/home/vietdu91/42_works/minishell
\$> unset HOME \$> export HOME \$> cd					ok	\$>	1	
\$> cd minishell Docs crashtest.c					ok	bash: cd: too many arguments \$>	1	
\$> cd /   echo \$? \$> pwd					ok	0 \$>	0 0	/home/vietdu91/42_works/minishell
\$> cd ~					ok	\$>	0	/home/vietdu91
\$> cd ~/	NON GERE				ok	\$>	0	/home/vietdu91
\$> cd ~/   echo \$? \$> pwd	NON GERE					0 \$>	0	/home/vietdu91/42_works/minishell
\$> cd *	NON GERE					\$>	0	[BONUS] S'il n'y a qu'un seul dossier et rien d'autre
\$> cd *	NON GERE					bash: cd: minishell: No such file or directory \$>	1	[BONUS] S'il n'y a qu'un seul fichier et rien d'autre
\$> cd *	NON GERE					bash: cd: too many arguments \$>	1	[BONUS] S'il y au moins deux éléments
\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r .././a \$> cd ..					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	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	/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	/home \$> /home/vietdu91/42_works/minishell/a/b \$>	0 0	
\$> mkdir a \$> cd a \$> rm -r ../a \$> echo \$PWD \$> echo \$OLDPWD					OK	../../../../home/vietdu91/42_works/minishell/a \$> ../../../../home/vietdu91/42_works/minishell \$>	0 0	
\$> export CDPATH= \$> cd \$HOME/..	NON GERE					\$>	0 0	A gerer ? /home
\$> export CDPATH= \$> cd home/vietdu91	NON GERE					/home/vietdu91 \$>	0 0	A gerer ? /home/vietdu91
\$> export CDPATH=./ \$> cd .	NON GERE					\$>	0 0	A gerer ? /home/vietdu91/42_works/minishell
\$> export CDPATH=./ \$> cd ..	NON GERE					\$>	0 0	A gerer ? /home/vietdu91/42_works
\$> chmod 000 minishell \$> ./minishell					ok	bash: ./minishell : Permission denied \$>	126	
\$> ls hola					ok	ls: cannot access 'hola': No such file or directory \$>	2	
\$> ./Makefile					ok	bash: ./Makefile: Permission denied \$>	126	

BÂTARDS	\$> ./minishell				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				\$>	0 0 0 0 0	
	\$> touch hola \$> ./hola				ok	bash: ./hola: Permission denied \$>	126	
	\$> env  "wc" -l				ok	66 \$>	0	
	\$> env  "wc " -l				ok	bash: wc -l: command not found \$>	127	
	\$> expr 1 + 1				ok	2 \$>	0	
	\$> expr \$? + \$?				ok	0 \$>	0	Fait la somme de la valeur de \$?
	\$> env -i ./minishell \$> env				pas gerer	PWD=/mnt/nfs/homes/emtran/Desktop/42_minishell SHLVL=1 _=/usr/bin/env \$>	0	
	\$> env -i ./minishell \$> export				pas gerer	export OLDPWD export PWD="/mnt/nfs/homes/emtran/Desktop/42_minishell" export SHLVL="1" \$>	0	
	\$> env -i ./minishell \$> cd				KO	bash: cd: HOME not set \$>	1	
	\$> env -i ./minishell \$> cd ~	NON GERE			KO	\$>	0	/home/vietdu91
	\$> exit				ok	exit ~/minishell \$>	0	
	\$> exit exit				ok	exit bash: exit: exit: numeric argument required ~/minishell \$>	2	
	\$> exit hola				ok	exit bash: exit: hola: numeric argument required ~/minishell \$>	2	
	\$> exit hola que tal				ok	exit bash: exit: hola: numeric argument required ~/minishell \$>	2	
	\$> exit 42				ok	exit ~/minishell \$>	42	
	\$> exit 000042				ok	exit ~/minishell \$>	42	
	\$> exit 666				ok	exit ~/minishell \$>	154	
	\$> exit 666 666				ok	exit bash: exit: too many arguments \$>	1	
	\$> exit -666 666				ok	exit bash: exit: too many arguments \$>	1	
	\$> exit hola 666				ok	exit bash: exit: hola: numeric argument required ~/minishell \$>	2	
	\$> exit 666 666 666 666				ok	exit bash: exit: too many arguments \$>	1	
	\$> exit 666 hola 666				ok	exit bash: exit: too many arguments \$>	1	
	\$> exit hola 666 666				OK	exit bash: exit: hola: numeric argument required ~/minishell \$>	2	
	\$> exit 259				ok	exit ~/minishell \$>	3	
	\$> exit -4				ok	exit ~/minishell \$>	252	
	\$> exit -42				ok	exit ~/minishell \$>	214	
	\$> exit -0000042				ok	exit ~/minishell \$>	214	
	\$> exit -259				ok	exit ~/minishell \$>	253	

EXIT	\$> exit -666				ok	exit ~/minishell \$>	102		
	\$> exit +666				ok	exit ~/minishell \$>	154		
	\$> exit 0				ok	exit ~/minishell \$>	0		
	\$> exit +0				ok	exit ~/minishell \$>	0		
	\$> exit -0				ok	exit ~/minishell \$>	0		
	\$> exit +42				ok	exit ~/minishell \$>	42		
	\$> exit -69 -96				ok	exit bash: exit: too many arguments \$>	1		
	\$> exit --666				OK	exit bash: exit: --666: numeric argument required ~/minishell \$>	2		
	\$> exit ++++666				ok	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2		
	\$> exit ++++++0				ok	exit bash: exit: ++++++0: numeric argument required ~/minishell \$>	2		
	\$> exit -----0				ok	exit bash: exit: -----0: numeric argument required ~/minishell \$>	2		
	\$> exit "666"				ok	exit ~/minishell \$>	154		
	\$> exit '666'				ok	exit ~/minishell \$>	154		
	\$> exit '-666'				ok	exit ~/minishell \$>	102		Marche aussi pour les double quotes
	\$> exit '+666'				ok	exit ~/minishell \$>	154		Marche aussi pour les double quotes
	\$> exit '----666'				ok	exit bash: exit: ----666: numeric argument required ~/minishell \$>	2		Marche aussi pour les double quotes
	\$> exit '++++666'				ok	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2		Marche aussi pour les double quotes
	\$> exit '6'66				ok	exit ~/minishell \$>	154		Marche aussi pour les double quotes
	\$> exit '2'66'32'				ok	exit ~/minishell \$>	8		Marche aussi pour les double quotes
	\$> exit "'666'"				ok	exit bash: exit: '666': numeric argument required ~/minishell \$>	2		
	\$> exit '"666"'				ok	exit bash: exit: "666": numeric argument required ~/minishell \$>	2		
	\$> exit '666'"666"666				ok	exit ~/minishell \$>	170		
	\$> exit +'666'"666"666				ok	exit ~/minishell \$>	170		
	\$> exit -'666'"666"666				ok	exit ~/minishell \$>	86		
	\$> exit 9223372036854775807				ok	exit ~/minishell \$>	255		
	\$> exit 9223372036854775808				ok	exit bash: exit: 9223372036854775808: numeric argument required ~/minishell \$>	2		
	\$> exit -9223372036854775808				ok	exit ~/minishell \$>	0		
	\$> exit -9223372036854775809				ok	exit bash: exit: -9223372036854775809: numeric argument required ~/minishell \$>	2		
	\$> cat   cat   cat   ls				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



PIPES	\$> ls   exit			leak	ok	\$>	0	Ne fait rien
	\$> ls   exit 42			leak	ok / KO data->status	\$>	42	Ne fait rien
	\$> exit   ls			leak	ok	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	
	\$> echo hola > bonjour			leak	ok	hola\$ \$>	0	
	\$> exit   cat -e bonjour			leak	ok	\$>	0	
	\$> echo hola > bonjour			leak	ok	\$>	0	
	\$> cat -e bonjour   exit			leak	ok	\$>	0	
	\$> echo   echo			leak	ok	\$>	0	
	\$> echo hola   echo que tal			leak	ok	que tal \$>	0	
	\$> pwd   echo hola			leak	ok	hola \$>	0	
	\$> env   echo hola			leak	ok	hola \$>	0	
	\$> echo oui   cat -e			leak	ok	oui\$ \$>	0	
	\$> echo oui   echo non   echo hola   grep oui			leak	ok	\$>	0	
	\$> echo oui   echo non   echo hola   grep non			leak	ok	\$>	0	
	\$> echo oui   echo non   echo hola   grep hola			leak	ok	hola \$>	0	
	\$> echo hola   cat -e   cat -e   cat -e			leak	ok	hola\$\$\$ \$>	0	
	\$> cd ..   echo "hola"			leak	ok	hola \$>	0	/home/vietdu91/42_works/minishell
	\$> cd /   echo "hola"			leak	ok	hola \$>	0	/home/vietdu91/42_works/minishell
	\$> cd ..   pwd			leak cd + bu	ok	/home/vietdu91/42_works/minishell \$>	0	/home/vietdu91/42_works/minishell
	\$> ifconfig   grep ":"				ok	docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500 ether 02:42:27:cf:95:49 txqueuelen 0 (Ethernet) [...] \$>	0	
	\$> ifconfig   grep hola				ok	\$>	1	
	\$> whoami   grep \$USER				ok	vietdu91 \$>	0	
	\$> whoami   grep \$USER > /tmp/bonjour				ok	vietdu91 \$>	0	
	\$> cat /tmp/bonjour				ok	\$>	0	
	\$> whoami   cat -e   cat -e > /tmp/bonjour				ok	vietdu91\$\$ \$>	0	
	\$> cat /tmp/bonjour				ok	\$>	0	
	\$> whereis ls   cat -e   cat -e > /tmp/bonjour				ok	ls: /usr/bin/ls /usr/share/man/man1/ls.1posix.gz /usr/share/man/man1/ls.1.gz\$\$ \$>	0	
	\$> cat /tmp/bonjour				ok	\$>	0	
	\$> ls   hola				ok	bash: hola: command not found \$>	127	
	\$> ls   ls hola				ok	ls: cannot access 'hola': No such file or directory \$>	2	
	\$> ls   ls   hola				ok	bash: hola: command not found \$>	127	
	\$> ls   hola   ls				ok	bash: hola: command not found crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	
	\$> ls   ls   hola   rev				ok	bash: hola: command not found \$>	0	
	\$> ls   ls   echo hola   rev			leak	ok	aloh \$>	0	
	\$> ls -la   grep "."				ok	total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root [...] .. -rwxr-xr-x 8 vietdu91 [...] crashtest.c [...] \$>	0	Sort la commande entiere de ls -la
	\$> ls -la   grep "'.'"				ok	\$>	0	

[illegible]

= &&	\$> echo -n bonjour && echo -n hola	NON GERE				bonjourhola\$>	0	[BONUS]
	\$> pwd && ls && echo hola	NON GERE				/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				/home/vietdu91/42_works/minishell hola \$>	0	[BONUS]
	\$> pwd && ls    echo hola	NON GERE				/home/vietdu91/42_works/minishell crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> pwd    ls    echo hola	NON GERE				/home/vietdu91/42_works/minishell \$>	0	[BONUS]
	\$> ls    export ""	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> export ""    ls	NON GERE				bash: export: `': not a valid identifier crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> ls && export ""	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c bash: export: `': not a valid identifier \$>	1	[BONUS]
	\$> export "" && ls	NON GERE				bash: export: `': not a valid identifier \$>	1	[BONUS]
	\$> ls    ;	NON GERE				bash: syntax error near unexpected token `;' \$>	2	[BONUS] Marche pour les metacharacters  &();)
	\$> ;    ls	NON GERE				bash: syntax error near unexpected token `;' \$>	2	[BONUS] Marche pour les metacharacters  &();)
	\$> ls && ;	NON GERE				bash: syntax error near unexpected token `;' \$>	2	[BONUS] Marche pour les metacharacters  &();)
	\$> ; && ls	NON GERE				bash: syntax error near unexpected token `;' \$>	2	[BONUS] Marche pour les metacharacters  &();)
	\$> ls    <	NON GERE				bash: syntax error near unexpected token `newline' \$>	2	[BONUS] Marche pour les metacharacters <>
	\$> ls && <	NON GERE				bash: syntax error near unexpected token `newline' \$>	2	[BONUS] Marche pour les metacharacters <>
	\$> cat   echo    ls	NON GERE				\$>	0	[BONUS]
	\$> cat   echo && ls	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> ls    cat   echo	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> ls && cat   echo	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> export "" && unset ""	NON GERE				bash: export: `': not a valid identifier	1	[BONUS]
	\$> (ls)	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> ( ( ls ) )	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> ( ( ) ls )	NON GERE				bash: syntax error near unexpected token `)'' \$>	2	[BONUS]
	\$> ls && (ls)	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]

( PARENTHESES )	<pre>\$&gt; (ls &amp;&amp; pwd)</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; ( ( ls&amp;&amp;pwd ) )</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; ( ( ls ) &amp;&amp;pwd )</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; (ls &amp;&amp; ( ( pwd ) ) )</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; (ls &amp;&amp; pwd) &gt; hola \$&gt; cat hola</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; &gt; hola ls &amp;&amp; pwd</pre>	NON GERE				<pre>/home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; &gt; hola (ls &amp;&amp; pwd)</pre>	NON GERE				<pre>bash: syntax error near unexpected token `(' \$&gt;</pre>	2		[BONUS]
	<pre>\$&gt; (&gt; pwd) \$&gt; ls</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; (&lt; pwd) \$&gt; ls</pre>	NON GERE				<pre>\$&gt;</pre>	0		[BONUS] Si pwd existe
	<pre>\$&gt; (&lt; pwd)</pre>	NON GERE				<pre>bash: pwd: No such file or directory \$&gt;</pre>	1		[BONUS] Si pwd n'existe pas
	<pre>\$&gt; ( ( ( ( ( pwd ) ) ) ) )</pre>	NON GERE				<pre>/home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; () pwd</pre>	NON GERE				<pre>bash: syntax error near unexpected token `)' \$&gt;</pre>	2		[BONUS]
	<pre>\$&gt; &gt; pwd (ls)</pre>	NON GERE				<pre>bash: syntax error near unexpected token `(' \$&gt;</pre>	2		[BONUS]
	<pre>\$&gt; (ls  pwd)&amp;&amp;(ls  pwd)</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; (lss  pwd)&amp;&amp;(lss  pwd)</pre>	NON GERE				<pre>bash: lss: command not found /home/vietdu91/42_works/minishell bash: lss: command not found /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
	<pre>\$&gt; (lss&amp;&amp;pwd)&amp;&amp;(lss&amp;&amp;pwd)</pre>	NON GERE				<pre>bash: lss: command not found \$&gt;</pre>	127		[BONUS]
	<pre>\$&gt; (ls &amp;&amp; pwd   wc) &gt; hola \$&gt; cat hola</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c 1 1 34 \$&gt;</pre>	0 0		[BONUS]
	<pre>\$&gt; (ls &amp;&amp; pwd   wc) &gt; hola \$&gt; (ls &amp;&amp; pwd   wc) &gt; hola \$&gt; cat hola</pre>	NON GERE				<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c 1 1 34 \$&gt;</pre>	0 0 0		[BONUS]

<pre>\$&gt; (ls &amp;&amp; pwd   wc) &gt;&gt; hola \$&gt; echo hey&amp;&amp;(ls &amp;&amp; pwd   wc) &gt; hola \$&gt; cat hola</pre>	NON GERE					<pre>hey \$&gt; crashtest.c Docs [...] srcs test.c       1      1      34 crashtest.c Docs [...] srcs test.c       1      1      34 \$&gt;</pre>	0 0 0		[BONUS]
<pre>\$&gt; (pwd   wc) &lt; hola</pre>	NON GERE					<pre>      1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; (ls &amp;&amp; pwd   wc) &lt; hola</pre>	NON GERE					<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c       1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; (ls -z    pwd   wc) &lt; hola</pre>	NON GERE					<pre>ls: invalid option -- 'z'       1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; echo hey &gt; hola \$&gt; (pwd   wc) &lt; hola</pre>	NON GERE					<pre>      1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; echo hey &gt; hola \$&gt; (ls &amp;&amp; pwd   wc) &lt; hola</pre>	NON GERE					<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c       1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; echo hey &gt; hola \$&gt; (ls -z    pwd   wc) &lt; hola</pre>	NON GERE					<pre>ls: invalid option -- 'z'       1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; (ls -z    pwd &amp;&amp; ls)</pre>	NON GERE					<pre>ls: invalid option -- 'z' /home/vietdu91/42_works/minishell crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; ls    (cat Makefile grep srcs) &amp;&amp; (pwd wc)</pre>	NON GERE					<pre>crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c       1      1      34 \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; ls -z &amp;&amp; (ls) &amp;&amp; (pwd)</pre>	NON GERE					<pre>ls: invalid option -- 'z' \$&gt;</pre>	130		[BONUS]
<pre>\$&gt; (ls &gt; Docs/hey &amp;&amp; pwd) &gt; hola \$&gt; cat hola \$&gt; cat Docs/hey</pre>	NON GERE					<pre>/home/vietdu91/42_works/minishell \$&gt; crashtest.c Docs [...] srcs test.c \$&gt;</pre>	0 0 0		[BONUS]
<pre>\$&gt; ls &gt; Docs/hey &amp;&amp; pwd &gt; hola \$&gt; cat hola \$&gt; cat Docs/hey</pre>	NON GERE					<pre>/home/vietdu91/42_works/minishell \$&gt; crashtest.c Docs [...] srcs test.c \$&gt;</pre>	0 0 0		[BONUS]
<pre>\$&gt; cd ../../ &amp;&amp; pwd &amp;&amp; pwd</pre>	NON GERE					<pre>/home/vietdu91/ /home/vietdu91/ \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; (cd ../../ &amp;&amp; pwd) &amp;&amp; pwd</pre>	NON GERE					<pre>/home/vietdu91/ /home/vietdu91/42_works/minishell \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; ls -z    cd ../../.&amp;&amp;pwd \$&gt; pwd</pre>	NON GERE					<pre>ls: invalid option -- 'z' /home/vietdu91 \$&gt; /home/vietdu91 \$&gt;</pre>	0 0		[BONUS]
<pre>\$&gt; ls -z    (cd ../../.&amp;&amp;pwd) \$&gt; pwd</pre>	NON GERE					<pre>ls: invalid option -- 'z' /home/vietdu91 \$&gt; /home/vietdu91/42_works/minishell \$&gt;</pre>	0 0		[BONUS]
<pre>\$&gt; echo hola &gt; bonjour \$&gt; cat bonjour</pre>					ok	<pre>hola \$&gt;</pre>	0 0		

<pre>\$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat bonjour</pre>					ok	<pre>hola que tal \$&gt;</pre>	0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat &lt; bonjour</pre>					ok	<pre>\$&gt; \$&gt; hola que tal \$&gt;</pre>	0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; rm bonjour \$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat &lt; bonjour</pre>					ok	<pre>que tal \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; echo hola que tal &gt; bonjour \$&gt; cat bonjour</pre>					ok	<pre>hola que tal \$&gt;</pre>	0 0		
<pre>\$&gt; echo hola que tal &gt; /tmp/bonjour \$&gt; cat -e /tmp/bonjour</pre>					ok	<pre>hola que tal\$ \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=hey \$&gt; echo bonjour &gt; \$HOLA \$&gt; echo \$HOLA</pre>					ok	<pre>\$&gt; hey \$&gt;</pre>	0 0 0		Ne fait rien
<pre>\$&gt; whereis grep &gt; Docs/bonjour \$&gt; cat Docs/bonjour</pre>					ok	<pre>grep: /usr/bin/grep /usr/share/man/man1/grep.1.gz /usr/share/man/man1/grep.1posix.gz /usr/share/info/grep. info.gz \$&gt;</pre>	0 0		
<pre>\$&gt; ls -la &gt; Docs/bonjour \$&gt; cat Docs/bonjour</pre>					ok	<pre>total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root   [...] .. -rwxr-xr-x  8 vietdu91 [...] crashtest.c [...] \$&gt;</pre>	0 0		
<pre>\$&gt; pwd&gt;bonjour \$&gt; cat bonjour</pre>					ok	<pre>/home/vietdu91/42_works/minishell \$&gt;</pre>	0 0		
<pre>\$&gt; pwd &gt;                bonjour \$&gt; cat bonjour</pre>					ok	<pre>/home/vietdu91/42_works/minishell \$&gt;</pre>	0 0		
<pre>\$&gt; echo hola &gt; &gt; bonjour \$&gt;</pre>					ok	<pre>bash: syntax error near unexpected token `&gt;' \$&gt;</pre>	2		
<pre>\$&gt; echo hola &lt; &lt; bonjour \$&gt;</pre>					ok	<pre>bash: syntax error near unexpected token `&lt;' \$&gt;</pre>	2		
<pre>\$&gt; echo hola &gt;&gt;&gt; bonjour \$&gt;</pre>					ok	<pre>bash: syntax error near unexpected token `&gt;' \$&gt;</pre>	2		
<pre>\$&gt; &gt; bonjour echo hola \$&gt; cat bonjour</pre>					ok	<pre>hola \$&gt;</pre>	0 0		
<pre>\$&gt; &gt; bonjour   echo hola \$&gt; cat bonjour</pre>					ok	<pre>hola \$&gt; \$&gt;</pre>	0 0 0		
<pre>\$&gt; prout hola &gt; bonjour \$&gt; ls</pre>					ok	<pre>bash: prout: command not found \$&gt; bonjour [...]</pre>	127 0		Erreur mais file "bonjour" cree tout de meme
<pre>\$&gt; echo hola &gt; hello &gt;&gt; hello &gt;&gt; hello \$&gt; ls \$&gt; cat hello</pre>					ok	<pre>\$&gt; hello \$&gt; hola \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; echo hola &gt; hello &gt;&gt; hello &gt;&gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; cat &lt; hello</pre>					ok	<pre>hola hola \$&gt;</pre>	0 0 0		
<pre>\$&gt; echo hola &gt; hello &gt;&gt; hello &gt;&gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; echo hola &gt; hello &gt;&gt; hello &gt;&gt; hello \$&gt; cat &lt; hello</pre>					ok	<pre>hola \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt;&gt;&gt; hello &gt;&gt; hello &gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; cat &lt; hello</pre>					ok	<pre>hola hola \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; echo hola &gt;&gt;&gt; hello &gt;&gt; hello &gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; echo hola &gt;&gt; hello &gt;&gt; hello &gt; hello \$&gt; cat &lt; hello</pre>					ok	<pre>hola \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; hello \$&gt; echo hola &gt;&gt; hello &gt;&gt; hello &gt;&gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; cat &lt; hello</pre>					ok	<pre>hola hola hola \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; hello \$&gt; echo hey &gt; bonjour \$&gt; echo &lt;bonjour &lt;hello</pre>					ok	<pre>\$&gt;</pre>	0 0 0		
<pre>\$&gt; echo hola &gt; hello \$&gt; echo hey &gt; bonjour \$&gt; echo &lt;hello &lt;bonjour</pre>					ok	<pre>\$&gt;</pre>	0 0 0		

<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; echo hola &gt; bonjour &gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>					ok	<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; echo hola &gt; bonjour &gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; echo hola &gt; bonjour &gt;&gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; echo hola &gt; bonjour &gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; echo hola &gt; bonjour &gt; hello &gt;&gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; echo hola &gt; bonjour &gt; hello &gt;&gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; echo hola &gt;&gt; bonjour &gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; echo hola &gt;&gt; bonjour &gt; hello &gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; echo hola &gt;&gt; bonjour &gt;&gt; hello &gt;&gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola \$&gt; \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; echo hola &gt;&gt; bonjour &gt;&gt; hello &gt;&gt; bonjour \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>hola hola \$&gt; hey \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; &gt; bonjour echo hola bonjour \$&gt; cat bonjour</pre>						<pre>bonjour hola \$&gt;</pre>	0 0		
<pre>\$&gt; &gt;bonjour echo &gt; hola&gt;bonjour&gt;hola&gt;&gt;bonjour&gt;hola hey &gt;bonjour hola &gt;hola \$&gt; cat bonjour \$&gt; cat hola</pre>						<pre>\$&gt; \$&gt; hey hola \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; echo bonjour &gt; hola1 \$&gt; echo hello &gt; hola2 \$&gt; echo 2 &gt;hola1 &gt;&gt; hola2 \$&gt; ls \$&gt; cat hola1 \$&gt; cat hola2</pre>						<pre>hola1 hola2 [...] \$&gt; \$&gt; hello 2 \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; echo bonjour &gt; hola1 \$&gt; echo hello &gt; hola2 \$&gt; echo 2 &gt;&gt;hola1 &gt; hola2 \$&gt; ls \$&gt; cat hola1 \$&gt; cat hola2</pre>						<pre>hola1 hola2 [...] \$&gt; bonjour \$&gt; 2 \$&gt;</pre>	0 0 0 0 0 0		
<pre>\$&gt; &gt; pwd \$&gt; ls</pre>					ok	<pre>pwd \$&gt;</pre>	0 0		
<pre>\$&gt; &lt; pwd</pre>					ok	<pre>bash: pwd: No such file or directory \$&gt;</pre>	1		





❖ ❖ ❖ ❖ ❖	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;srcs/bonjour &lt;prout &gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; rm srcs/bonjour srcs/hello \$&gt; &gt;srcs/bonjour &lt;prout &gt;srcs/hello \$&gt; ls srcs \$&gt; cat srcs/bonjour</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; \$&gt;</pre>	0 0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; ../bonjour \$&gt; echo hey &gt; ../hello \$&gt; &gt;../bonjour &gt;../hello &lt;prout \$&gt; cat ../bonjour ../hello</pre>					<pre>bash: prout: No such file or directory \$&gt; \$&gt;</pre>	0 0 1 0		
	<pre>\$&gt; echo hola &gt; ../bonjour \$&gt; echo hey &gt; ../hello \$&gt; rm ../bonjour ../hello \$&gt; &gt;../bonjour &gt;../hello &lt;prout \$&gt; ls .. \$&gt; cat ../bonjour ../hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour  hello \$&gt; \$&gt;</pre>	0 0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; ../bonjour \$&gt; echo hey &gt; ../hello \$&gt; &gt;../bonjour &lt;prout &gt;../hello \$&gt; cat ../bonjour \$&gt; cat ../hello</pre>					<pre>bash: prout: No such file or directory \$&gt; \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; ../bonjour \$&gt; echo hey &gt; ../hello \$&gt; rm ../bonjour ../hello \$&gt; &gt;../bonjour &lt;prout &gt;../hello \$&gt; ls .. \$&gt; cat ../bonjour</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; \$&gt;</pre>	0 0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;srcs/bonjour &gt;&gt;srcs/hello &lt;prout \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour &gt;srcs/hello &lt;prout \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour &gt;&gt;srcs/hello &lt;prout \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;srcs/bonjour &lt;prout &gt;&gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour &lt;prout &gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour &lt;prout &gt;&gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &lt;prout &gt;&gt;srcs/bonjour &gt;&gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</pre>					<pre>bash: prout: No such file or directory \$&gt; bonjour \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
	<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; &lt;bonjour &gt;hello \$&gt; cat bonjour \$&gt; cat hello</pre>					<pre>hola \$&gt; &gt;&gt; \$&gt; \$&gt;</pre>	0 0 0 0 0		

<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; &gt;bonjour &gt;hello &lt; prout \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>bash: prout: No such file or directory \$&gt; \$&gt; \$&gt;</pre>	0 0 1 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; &gt;bonjour &gt;hello &lt; prout \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>bash: prout: No such file or directory \$&gt; \$&gt; \$&gt; \$&gt;</pre>	0 0 0 1 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; &gt;bonjour &lt;prout hello \$&gt; cat bonjour \$&gt; cat hello</pre>						<pre>bash: prout: No such file or directory \$&gt; \$&gt; \$&gt; hey \$&gt;</pre>	0 0 1 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; rm bonjour hello \$&gt; &gt;bonjour &lt;prout hello \$&gt; cat bonjour</pre>						<pre>bash: prout: No such file or directory \$&gt; \$&gt;</pre>	0 0 0 1 0		Le fichier hello n'existe pas
<pre>\$&gt; echo hola &gt; bonjour \$&gt; &lt;bonjour cat   wc &gt; bonjour \$&gt; cat bonjour</pre>						<pre>0 0 0 \$&gt;</pre>	0 0 0		
<pre>\$&gt; rm -f bonjour \$&gt; rm bonjour &gt; bonjour \$&gt; ls -l bonjour</pre>						<pre>ls: cannot access 'bonjour': No such file or directory \$&gt;</pre>	0 0 2		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA \$&gt; ls</pre>						<pre>'bonjour hello' [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;"\$HOLA" \$&gt; ls</pre>						<pre>'bonjour hello' [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$"HOLA" \$&gt; ls</pre>						<pre>HOLA [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA&gt;hey \$&gt; ls</pre>						<pre>'bonjour hello' hey [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;hey&gt;\$HOLA \$&gt; ls</pre>						<pre>'bonjour hello' hey [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;hey&gt;\$HOLA&gt;hey&gt;hey \$&gt; ls</pre>						<pre>'bonjour hello' hey [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export A=hey \$&gt; export A B=HOLA D E C="Que Tal" \$&gt; echo \$PROUT\$B\$C &gt; /tmp/a &gt; /tmp/b &gt; /tmp/c \$&gt; cat /tmp/a \$&gt; cat /tmp/b \$&gt; cat /tmp/c</pre>						<pre>\$&gt; \$&gt; \$&gt; \$&gt; HolaQue Tal \$&gt;</pre>	0 0 0 0 0 0		
<pre>!!!! Contenu du fichier a : "Amour Tu es Horrible" !!!! Contenu du fichier b : "0123456789" !!!! Contenu du fichier c : "Prout"</pre>									
<pre>\$&gt; &lt;a cat &lt;b &lt;c</pre>						<pre>Prout \$&gt;</pre>	0		
<pre>\$&gt; &lt;a cat &lt;b &lt;c \$&gt; cat a \$&gt; cat b \$&gt; cat c</pre>						<pre>\$&gt; \$&gt; \$&gt; \$&gt; Amour Tu es Horrible \$&gt; 0123456789 \$&gt; Prout \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; &gt;a ls &gt;b &gt;&gt;c &gt;d \$&gt; cat a \$&gt; cat b \$&gt; cat c \$&gt; cat d</pre>						<pre>\$&gt; \$&gt; \$&gt; \$&gt; Prout \$&gt; \$&gt; crashtest.c \$&gt; Docs \$&gt; [...] \$&gt; test.c \$&gt;</pre>	0 0 0 0 0 0		

	<pre>\$&gt; &gt;a ls &gt;b &gt;&gt;c &gt;d \$&gt; cat a \$&gt; cat b \$&gt; cat c \$&gt; cat d</pre>					<pre>\$&gt; \$&gt; \$&gt; Prout \$&gt; crashtest.c Docs [...] test.c \$&gt;</pre>	0 0 0 0 0		
	<pre>\$&gt; echo hola &gt; a &gt; b &gt; c \$&gt; cat a \$&gt; cat b \$&gt; cat c</pre>					<pre>\$&gt; \$&gt; hola \$&gt;</pre>	0		
	<pre>\$&gt; mkdir dir \$&gt; ls -la &gt; dir/bonjour \$&gt; cat dir/bonjour</pre>					<pre>total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root    [...] .. [...] \$&gt;</pre>	0		
	<pre>\$&gt; &lt;a \$&gt; cat a</pre>					<pre>Amour Tu es Horrible \$&gt;</pre>	0		
	<pre>\$&gt; &gt;d cat &lt;a &gt;&gt;e \$&gt; cat a \$&gt; cat d \$&gt; cat e</pre>					<pre>\$&gt; Amour \$&gt; Amour \$&gt;</pre>	0 0 0 0		
	<pre>\$&gt; &lt; a &gt; b cat &gt; hey &gt;&gt; d \$&gt; cat d \$&gt; ls</pre>					<pre>Amour Tu Es Horrible \$&gt; [...] a b hey d</pre>	0 0 0		
HEREDOC	<pre>\$&gt; cat &lt;&lt; hola</pre>				ok	<pre>&gt; \$HOME &gt; hola /home/vietdu91 \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt; 'hola'</pre>				ko	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt; "hola"</pre>				ko	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt; ho"la"</pre>				ko	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt; \$HOME</pre>				ok	<pre>&gt; prout &gt; /home/vietdu91 &gt; \$HO &gt; \$HOME \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt; hola &gt; bonjour \$&gt; cat bonjour</pre>				ok	<pre>&gt; prout &gt; croutte &gt; antoine pue &gt; hola \$&gt; prout croutte antoine pue</pre>	0		
	<pre>\$&gt; cat &lt;&lt; hola   rev</pre>				ok	<pre>&gt; prout &gt; croutte &gt; antoine pue &gt; hola tuorp ettuorc eup eniotna \$&gt;</pre>	0		
	<pre>\$&gt; &lt;&lt; hola</pre>				ok	<pre>&gt; chola &gt; holache &gt;      hola &gt; "hola" &gt; 'hola' &gt; hola \$&gt;</pre>	0		
	<pre>\$&gt; &lt;&lt;hola</pre>				ok	<pre>&gt; &gt; &gt; hola \$&gt;</pre>	0		
	<pre>\$&gt; cat &lt;&lt;</pre>				ok	<pre>bash: syntax error near unexpected token `newline' \$&gt;</pre>	2		

WILDCARD	\$> cat << prout << lol << koala				ok	> prout > lol > koala \$>	0	
	\$> prout << lol << cat << koala				ok	> prout > lol > cat > koala bash: prout: command not found	127	
	\$> << \$hola				ok	> \$hola \$>	0	
	\$> << \$"hola"\$"b"				KO	> holab \$>	0	
	\$> << \$"\$hola"\$"\$b"				KO	> \$hola\$\$b \$>	0	
	\$> << ho\$la\$"a"\$"\$b"				KO	> hola\$a\$\$b > ho\$la\$a\$\$nb \$>	0	
	\$> echo hola <<< bonjour				ok pas agerer	hola \$>	0	
	\$> echo hola <<<< bonjour				KO	bash: syntax error near unexpected token `<' \$>	2	
	\$> echo hola <<<< bonjour				ok	bash: syntax error near unexpected token `<<' \$>	2	
WILDCARD	\$> ls *	NON GERE				crashtest.c Docs Dockers Drawings Makefile minishell philosophers test.c  Docs: identite.pdf loyer.pdf srcs: main.c utils.c sort.c \$>	0	[BONUS]
	\$> ls *.*	NON GERE				crashtest.c test.c \$>	0	[BONUS]
	\$> ls *.hola	NON GERE				ls: cannot access '*.hola' : No such file or directory \$>	2	[BONUS]
	\$> cat M*le	NON GERE				[affiche le Makefile] \$>	0	[BONUS]
	\$> cat M*ee	NON GERE				cat: 'M*ee': No such file or directory	1	[BONUS]
	\$> cat Make*file	NON GERE				[affiche le Makefile] \$>	0	[BONUS]