



**[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)



## Gestion des comportements

[illegible]

[illegible]

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

\$> echo       Hola				OK	OK	Hola \$>	0		
\$> echo    Hola       Que     Tal				OK	OK	Hola Que Tal \$>	0		
\$> echo       \n hola				OK	OK	n hola \$>	0		A gerer ?
\$> echo "               "   cat -e				OK	OK	\$ \$>	0		
\$> echo                 cat -e				OK	OK	\$ \$>	0		
\$> """"echo hola"""""""" que"""""" tal""""				OK	OK	hola que tal \$>	0		
\$> echo -n -n				OK	OK	\$>	0		
\$> echo -n -n Hola Que				OK	OK	Hola Que\$>	0		
\$> echo -p				OK	OK	-p \$>	0		
\$> echo -nnnnn				OK	OK	\$>	0		
\$> echo -n -nnn -nnnn				OK	OK	\$>	0		
\$> echo -n-nnn -nnnn				OK	OK	-n-nnn -nnnn \$>	0		
\$> echo -n -nnn hola -nnnn				OK	OK	hola -nnnn\$>	0		
\$> echo -n -nnn-nnnn				OK	OK	-nnn-nnnn\$>	0		
\$> echo -----n				OK	OK	-----n \$>	0		
\$> echo -nnn -----n				OK	OK	-----n\$>	0		
\$> echo -nnn -----nn---nnnn				OK	OK	-----nn---nnnn\$>	0		
\$> echo -nnn -----nnnn				OK	OK	-----nnnn\$>	0		
\$> echo \$				OK	OK	\$ \$>	0		
\$> echo \$?				OK	OK	0 \$>	0		Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
\$> echo \$?\$				OK	OK	0\$ \$>	0		
\$> echo \$?   echo \$?   echo \$?				OK	OK	0 \$>	0		
\$> echo \$:\$=   cat -e				OK	OK	:\$:\$= \$>	0		
\$> echo " \$ "   cat -e				OK	OK	\$ \$ \$>	0		
\$> echo ' \$ '   cat -e				OK	OK	\$ \$ \$>	0		
\$> echo \$HOME				OK	OK	/home/vietdu91 \$>	0		
\$> echo \HOME				OK	OK	\$HOME \$>	0		
\$> echo my shit terminal is [\$TERM]				OK	OK	my shit terminal is [xterm-256color] \$>	0		
\$> echo my shit terminal is [\$TERM4				OK	OK	my shit terminal is [ \$>	0		
\$> echo my shit terminal is [\$TERM4]				OK	OK	my shit terminal is [] \$>	0		
\$> echo \$UID				OK	OK	1000 \$>	0		
\$> echo \$HOME9				OK	OK	\$>	0		Marche pour tous les caractères alphanumériques de l'ASCII
\$> echo \$9HOME				OK	OK	HOME \$>	0		
\$> echo \$HOME%				OK	OK	/home/vietdu91% \$>	0		Marche pour les autres caractères (ponctuation, caractères non-ASCII, etc.)

<div>ECHO</div>	\$> 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 \$USER\$var\USER\$USER\USERtest\$USER			OK	OK	vietdu91\$USERvietdu91\$USERtestvietdu91 \$>	0	A ne pas gerer si '\ ' non gere : Doit sortir : vietdu91vietdu91vietdu91vietdu91
	\$> echo \$hola*			OK	OK	\$>	0	Cas d'un variable inexistant
	\$> echo -nnnn \$hola			OK	OK	\$>	0	
	\$> echo > <			OK	OK	bash: syntax error near unexpected token '<' \$>	2	
	\$> echo			OK	OK	bash: syntax error near unexpected token ` ' \$>	2	
	\$> EechoE			OK	OK	bash: EechoE: command not found \$>	127	
	\$> .echo.			OK	OK	bash: .echo.: command not found \$>	127	
	\$> >echo>			OK	OK	bash: syntax error near unexpected token `newline' \$>	2	
	\$> <echo<			OK	OK	bash: syntax error near unexpected token `newline' \$>	2	
	\$> >>echo>>			OK	OK	bash: syntax error near unexpected token `newline' \$>	2	
	\$>  echo			OK	OK	bash: syntax error near unexpected token ` ' \$>	2	
	\$>  echo -n hola			OK	OK	bash: syntax error near unexpected token ` ' \$>	2	
	\$> echo *	NON GERE		OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo '*'	NON GERE		OK	OK	* \$>	0	Marche aussi pour les double quotes
	\$> echo D*	NON GERE		OK	OK	Docs Dockers Drawings \$>	0	[BONUS]
	\$> echo *Z	NON GERE		OK	OK	*Z \$>	0	[BONUS]
	\$> echo *t hola	NON GERE		OK	OK	crashtest.c test.c hola \$>	0	[BONUS]
	\$> echo *t	NON GERE		OK	OK	crashtest.c test.c \$>	0	[BONUS]
	\$> echo \$*	NON GERE		OK	OK	\$>	0	[BONUS]
	\$> echo hola*hola *	NON GERE		OK	OK	hola*hola crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo \$hola*	NON GERE		OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> echo \$HOME*	NON GERE		OK	OK	/home/vietdu91 \$>	0	[BONUS]
	\$> echo \$""			OK	OK	\$>	0	
	\$> echo "\$"""			OK	OK	\$ \$>	0	
	\$> echo '\$'''			OK	OK	\$ \$>	0	
	\$> echo \$"HOME"			OK	OK	HOME \$>	0	
	\$> echo \$''HOME			OK	OK	HOME \$>	0	


\$> echo \$""HOME				OK	OK	HOME \$>	0		
\$> echo "\$H0"ME				OK	OK	ME \$>	0		
\$> echo '\$H0'ME				OK	OK	\$HOME \$>	0		
\$> echo "\$H0""ME"				OK	OK	ME \$>	0		
\$> echo '\$H0''ME'				OK	OK	\$HOME \$>	0		
\$> echo "'\$H0''ME'"				OK	OK	'ME' \$>	0		⚠ Il y a bien 3 singles quotes avant le ME. Je ne sais pas pourquoi Excel tronque à l'affichage le 3e quote
\$> echo ""\$HOME				OK	OK	/home/vietdu91 \$>	0		
\$> echo "" \$HOME				OK	OK	/home/vietdu91 \$>			
\$> echo ''\$HOME				OK	OK	/home/vietdu91 \$>	0		
\$> echo '' \$HOME				OK	OK	/home/vietdu91 \$>			
\$> echo \$"H0""ME"				OK	OK	HOME \$>	0		
\$> echo '\$H0''ME'				OK	OK	HOME \$>	0		
\$> echo '\$HOME'				OK	OK	HOME \$>	0		
\$> echo "\$"HOME				OK	OK	\$HOME \$>	0		
\$> echo \$=HOME				OK	OK	\$=HOME \$>	0		
\$> echo \$"HOLA"				OK	OK	HOLA \$>	0		
\$> echo '\$HOLA'				OK	OK	HOLA \$>	0		
\$> echo \$DONTEXIST Ho!a				OK	OK	Ho!a \$>	0		Le cas ou la variable n'existe pas et n'est pas importee
\$> echo "ho!a"				OK	OK	ho!a \$>	0		
\$> echo 'ho!a'				OK	OK	ho!a \$>	0		
\$> echo ''ho!a''				OK	OK	ho!a \$>	0		
\$> echo ''h'o'la''				OK	OK	ho!a \$>	0		
\$> echo ""'h'o'la'""				OK	OK	'h'o'la'' \$>	0		
\$> echo ""'h'o'la'""				OK	OK	ho!a' \$>	0		
\$> echo"ho!a'"				OK	OK	bash: echo'ho!a': command not found \$>	0		
\$> echo ""ho!a'""				OK	OK	'ho!a' \$>	0		
\$> echo ""ho!a'""				OK	OK	"ho!a" \$>	0		
\$> echo ''ho''''l'a''''				OK	OK	ho"l"a \$>	0		
\$> echo ho!a"*****"				OK	OK	ho!a \$>	0		

	\$> echo hola"			OK	OK	hola'	0	
	\$> echo hola'			OK	OK	hola	0	
	\$> echo hola"			OK	OK	hola"	0	
	\$> e"cho hola"			OK	OK	bash: echo hola: command not found	127	
	\$> e'cho hola'			OK	OK	bash: echo hola: command not found	127	
	\$> echo "hola "   cat -e			OK	OK	hola \$	0	
	\$> echo ""hola			OK	OK	hola	0	
	\$> echo "" hola			OK	OK	hola	0	
	\$> echo "" hola			OK	OK	hola	0	
	\$> echo "" hola			OK	OK	hola	0	
	\$> echo hola""bonjour			OK	OK	holabonjour	0	
	\$> "e"'c'ho 'b'"o"nj"o"'u'r			OK	OK	bonjour	0	
	\$> ""e"'c'ho 'b'"o"nj"o"'u'r"			OK	OK	bash: e'c'ho 'b'onjo'u'r: command not found	127	
	\$> echo "\$DONTEXIST"Makefile			OK	OK	Makefile	0	
	\$> echo "\$DONTEXIST""Makefile"			OK	OK	Makefile	0	
	\$> echo "\$DONTEXIST" "Makefile"			OK	OK	Makefile	0	Attention a l'espace avant
	\$> \$?			OK	OK	bash: 0: command not found	127	Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
	\$> \$? \$?			OK	OK	bash: 00: command not found	127	
	\$> ?\$HOME			OK	OK	bash: ?/home/vietdu91: command not found	127	
	\$> \$			OK	OK	bash: \$: command not found	127	
	\$> \$HOME			OK	OK	bash: /home/vietdu91: Is a directory	126	
	\$> \$HOME\$skjhfkds\$fsd			OK	OK	\$>	0	
	\$> "\$HOME\$skjhfkds\$fsd"			OK	OK	bash: : command not found	127	
	\$> '\$HOME\$skjhfkds\$fsd'			OK	OK	bash: \$HOME\$ddfdffssfd\$fs: command not found	127	
	\$> \$DONTEXIST			OK	OK	\$>	0	Cas d'un variable inexistant
	\$> \$LESS\$VAR			OK	OK	bash: -R: command not found	127	
	\$> Ctlr-C			OK	OK	\$> ^C	130	
	\$> holaCtlr-C			OK	OK	\$> hola^C	130	

SIGNALS ⚡	\$> cat (faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])			OK	OK	^C \$>	130	
	\$> cat   rev(faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])			OK	OK	^C \$>	0	
	\$> Ctlr-D			OK	OK	\$> exit ~/minishell \$>	0	Sort du programme
	\$> hola Ctlr-D			OK	OK	\$> hola	/	Ne fait rien
	\$> Ctlr-\			OK	OK	\$>	/	Ne fait rien
	\$> hola Ctlr-\			OK	OK	\$> hola	/	
	\$> cat (faire Ctlr-\ apres avoir fait plusieurs fois [ENTREE])			OK	OK	^\\Quit (core dumped) \$>	131	
	\$> Ctlr-Z			OK	OK	\$>	/	Ne fait rien
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-C une seconde apres)			OK	OK	^C \$>	130	
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-D une seconde apres)			OK	OK	\$>	0	Ne fait rien
	\$> sleep 3   sleep 3   sleep 3 (faire Ctlr-\ une seconde apres)			OK	OK	^\\Quit (core dumped) \$>	131	
	\$> env			OK	OK	... PATH=... \$>	0	
	\$> env hola	NON GERE		OK	OK	env: 'hola': No such file or directory \$>	127	
	\$> env hola que tal	NON GERE		OK	OK	env: 'hola': No such file or directory \$>	127	
	\$> env env	NON GERE		OK	OK	... PATH=... \$>	0	Affiche une seule fois env
	\$> env env env env env	NON GERE		OK	OK	... PATH=... \$>	0	Affiche une seule fois env
	\$> env ls	NON GERE		OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	
	\$> env ./Makefile	NON GERE		OK	OK	env: './Makefile': Permission denied \$>	126	
	\$> export HOLA=bonjour \$> env			OK	OK	... HOLA=bonjour \$>	0 0	
	\$> export HOLA=bonjour \$> env			OK	OK	... HOLA=bonjour \$>	0 0	
	\$> export			OK	OK	... export PATH="..." \$>	0	Les variables sont trieés par ordre alphabetique On n'oublie pas les doubles quotes qui designent les strings attribues a chaque variable
	\$> export Hola \$> export			OK	OK	... export HOME="/home/vietdu91" export Hola export PATH="..." \$>	0 0	Hola n'existe pas dans env mais il existe dans export




<pre>\$&gt; export Hola9hey \$&gt; export</pre>				OK	OK	<pre>... export HOME="/home/vietdu91" export Hola9hey export PATH="..." \$&gt;</pre>	0 0		
<pre>\$&gt; export \$DONTEXIST</pre>				OK	OK	<pre>... export HOME="/home/vietdu91" export PATH="..." \$&gt;</pre>	0		Le cas ou la variable n'existe pas et n'est pas importee
<pre>\$&gt; export   grep "HOME"</pre>				OK	OK	<pre>export HOME="/home/vietdu91" \$&gt;</pre>	0		
<pre>\$&gt; export ""</pre>				OK	OK	<pre>bash: export: `': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export =</pre>				OK	OK	<pre>bash: export: `=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export %</pre>				OK	OK	<pre>bash: export: `%': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export \$?</pre>				OK	OK	<pre>bash: export: `?': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export ?=2</pre>				OK	OK	<pre>bash: export: `?=2': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export 9HOLA=</pre>				OK	OK	<pre>bash: export: `9HOLA=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOLA9=bonjour \$&gt; env</pre>				OK	OK	<pre>... HOLA9=bonjour \$&gt;</pre>	0 0		
<pre>\$&gt; export _HOLA=bonjour \$&gt; env</pre>				OK	OK	<pre>... _HOLA=bonjour \$&gt;</pre>	0 0		
<pre>\$&gt; export __HOLA=bonjour \$&gt; env</pre>				OK	OK	<pre>... __HOLA=bonjour \$&gt;</pre>	0 0		
<pre>\$&gt; export _HO_LA=bonjour \$&gt; env</pre>				OK	OK	<pre>... _HO_LA=bonjour \$&gt;</pre>	0 0		
<pre>\$&gt; export HOL@=bonjour</pre>				OK	OK	<pre>bash: export: `HOL@=bonjour': not a valid identifier \$&gt;</pre>	1		Cas où le nom de variable ne contient pas de lettres, ni de chiffres, ni d'underscores
<pre>\$&gt; export HOL\~A=bonjour</pre>	NON GERE			OK	OK	<pre>bash: export: `HOL~A=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export -HOLA=bonjour</pre>				OK	OK	<pre>bash: export: -H: invalid option \$&gt;</pre>	2		
<pre>\$&gt; export --HOLA=bonjour</pre>				OK	OK	<pre>bash: export: --: invalid option \$&gt;</pre>	2		
<pre>\$&gt; export HOLA-=bonjour</pre>				OK	OK	<pre>bash: export: `HOLA-=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HO-LA=bonjour</pre>				OK	OK	<pre>bash: export: `HO-LA=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOL.A=bonjour</pre>				OK	OK	<pre>bash: export: `HOL.A=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOL\\\$A=bonjour</pre>	NON GERE			OK	OK	<pre>bash: export: `HOL\$A=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HO\\LA=bonjour</pre>	NON GERE			OK	OK	<pre>bash: export: `HO\\LA=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOL}A=bonjour</pre>				OK	OK	<pre>bash: export: `HOL}A=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOL{A=bonjour</pre>				OK	OK	<pre>bash: export: `HOL{A=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HO*LA=bonjour</pre>				OK	OK	<pre>bash: export: `HO*LA=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HO#LA=bonjour</pre>				OK	OK	<pre>bash: export: `HO#LA=bonjour': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HO@LA=bonjour</pre>				OK	OK	<pre>bash: export: `HO@LA=bonjour': not a valid identifier \$&gt;</pre>	1		

<div>  <div>ENV &amp; EXPORT &amp; UNSET</div>  </div>	\$> export H0!LA=bonjour			OK	OK	bash: !LA=bonjour: event not found	0	
	\$> export H0\$?LA=bonjour \$> env			OK	OK	... H0!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 \$> env	NON GERE		OK	OK	... HOLA=bonjour \$>	0	
	\$> export HOLA=bonjour \$> export HOLA+=bonjour \$> env	NON GERE		OK	OK	... HOLA=bonjourbonjour \$>	0	"+=" s'ajoute au contenu de la variable
	\$> exportHOLA=bonjour \$> env	NON GERE		OK	OK	... PATH=... \$>	0	
	\$> export HOLA =bonjour			OK	OK	bash: export: `=bonjour': not a valid identifier \$>	1	
	\$> export HOLA = bonjour			OK	OK	bash: export: `=': not a valid identifier \$>	1	
	\$> export HOLA=bon jour \$> env			OK	OK	... HOLA=bon \$>	0	
	\$> export HOLA= bonjour \$> env			OK	OK	... HOLA= \$>	0	
	\$> export HOLA=bonsoir \$> export HOLA=bonretour \$> export HOLA=bonjour \$> env			OK	OK	... HOLA=bonjour \$>	0 0 0	
	\$> export HOLA=\$HOME \$> env			OK	OK	... HOLA=/home/vietdu91 \$>	0 0	
	\$> export HOLA=bonjour\$HOME \$> env			OK	OK	... HOLA=bonjour/home/vietdu91 \$>	0 0	
	\$> export HOLA=\$HOMEbonjour \$> env			OK	OK	... HOLA= \$>	0 0	
	\$> export HOLA=bon\$jour \$> env			OK	OK	... HOLA=bon \$>	0 0	Le cas où la variable "\$jour" n'existe pas
	\$> export HOLA=bon\jour \$> env	NON GERE		OK	OK	... HOLA=bonjour \$>	0 0	
	\$> export HOLA=bon\\jour \$> env	NON GERE		OK	OK	... HOLA=bon\jour \$>	0 0	
	\$> export HOLA=bon(jour	NON GERE		OK	OK	bash: syntax error near unexpected token `('	2	Marche pour les caractères : ( )
	\$> export HOLA=bon()jour	NON GERE		OK	OK	bash: syntax error near unexpected token `('	2	
	\$> export HOLA=bon&jour	NON GERE		OK	OK	bash: jour: command not found \$>	127	Marche pour les caractères : &
	\$> export HOLA=bon@jour \$> env			OK	OK	\$> env ... HOLA=bon@jour \$>	0	

<pre>\$&gt; export HOLA=bon;jour \$&gt; env</pre>				OK	OK	<pre>bash: jour: command not found \$&gt; env ... HOLA=bon \$&gt;</pre>	127 0		
<pre>\$&gt; export HOLA=bon!jour</pre>				OK	OK	<pre>bash: !jour: event not found \$&gt;</pre>	1		
<pre>\$&gt; export HOLA=bon"jour" \$&gt; env</pre>				OK	OK	<pre>... HOLA=bonjour PATH=... \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA\$USER=bonjour \$&gt; env</pre>				OK	OK	<pre>... HOLAvietdu91=bonjour PATH=... \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=bonjour=casse-toi \$&gt; echo \$HOLA</pre>				OK	OK	<pre>HOLA=bonjour=casse-toi \$&gt;</pre>	0 0		
<pre>\$&gt; export "HOLA=bonjour"=casse-toi \$&gt; echo \$HOLA</pre>				OK	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	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	OK	<pre>bonjour et casse-toi \$&gt;</pre>	0 0		
<pre>\$&gt; export \$HOLA=bonjour \$&gt; env</pre>				OK	OK	<pre>... bonjour=bonjour PATH=... \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="bonjour      " \$&gt; echo \$HOLA   cat -e</pre>				OK	OK	<pre>bonjour\$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="      -n bonjour  " \$&gt; echo \$HOLA</pre>				OK	OK	<pre>bonjour\$&gt;</pre>	0		
<pre>\$&gt; export HOLA="bonjour  "/" \$&gt; echo \$HOLA</pre>				OK	OK	<pre>bonjour / \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="" \$&gt; echo " \$HOLA "   cat -e</pre>				OK	OK	<pre>" \$ \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=at \$&gt; c\$HOLA Makefile</pre>				OK	OK	<pre>[cela affiche le Makefile] \$&gt;</pre>	0		
<pre>\$&gt; export "" HOLA=bonjour \$&gt; env</pre>				OK	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	OK	<pre>cat Makefile   grep NAME \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=hey \$&gt; echo \$HOLA\$HOLA\$HOLA=hey\$HOLA</pre>				OK	OK	<pre>heyheyhey=heyhey \$&gt;</pre>	0		
<pre>\$&gt; export HOLA=" bonjour hey " \$&gt; echo \$HOLA   cat -e</pre>				OK	OK	<pre>bonjour hey\$ \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=" bonjour hey " \$&gt; echo """"\$HOLA""""   cat -e</pre>				OK	OK	<pre> bonjour hey \$ \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=" bonjour hey " \$&gt; echo wesh"\$HOLA"   cat -e</pre>				OK	OK	<pre>wesh bonjour hey \$ \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=" bonjour hey " \$&gt; echo wesh""\$HOLA.</pre>				OK	OK	<pre>wesh bonjour hey . \$&gt;</pre>	0 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	OK	<pre>weshHOLA. \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=" bonjour hey " \$&gt; echo wesh\$"HOLA HOLA".</pre>				OK	OK	<pre>weshHOLA HOLA. \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=" hola et \$HOLA" \$&gt; echo \$HOLA</pre>				OK	OK	<pre>hola et bonjour \$&gt;</pre>	0 0 0		

<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=' hola et \$HOLA' \$&gt; echo \$HOLA</pre>				OK	OK	<pre>hola et \$HOLA \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA=bonjour \$&gt; export HOLA=" hola et \$HOLA"\$HOLA \$&gt; echo \$HOLA</pre>				OK	OK	<pre>hola et bonjourbonjour \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="ls      -l      - a" \$&gt; echo \$HOLA</pre>				OK	OK	<pre>ls -l - a \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA="s -la" \$&gt; l\$HOLA</pre>				OK	OK	<pre>total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root    [...] .. -rwxr-xr-x  8 vietdu91 [...] crashtest.c [...] \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA="s -la" \$&gt; l"\$HOLA"</pre>				OK	OK	<pre>bash: ls -la: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA="s -la" \$&gt; l'\$HOLA'</pre>				OK	OK	<pre>bash: l\$HOLA: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA="l" \$&gt; \$HOLAs</pre>				OK	OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; export HOLA="l" \$&gt; "\$HOLA"s</pre>				OK	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	OK	<pre>... HOL=A=bonjour \$&gt;</pre>	0		
<pre>\$&gt; export HOLA="l" \$&gt; '\$HOLA's</pre>				OK	OK	<pre>bash: \$HOLAs: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOL=A="" \$&gt; env</pre>				OK	OK	<pre>... HOL=A= \$&gt;</pre>	0 0		
<pre>\$&gt; export TE+S=T \$&gt; env</pre>				OK	OK	<pre>bash: export: `TE+S=T': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export ""=""</pre>				OK	OK	<pre>bash: export: `=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export ''=''</pre>				OK	OK	<pre>bash: export: `=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export "="="="</pre>				OK	OK	<pre>bash: export: `===': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export '='='='</pre>				OK	OK	<pre>bash: export: `===': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; \$HOLA"BYE"d</pre>				OK	OK	<pre>bash: pBYEd: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; "\$HOLA"'\$BYE'd</pre>				OK	OK	<pre>bash: p\$BYEd: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; "\$HOLA""\$BYE"d</pre>				OK	OK	<pre>/home/vietdu91/42_works/minishell</pre>	0		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; \$"HOLA"\$"\$BYE"d</pre>				OK	OK	<pre>bash: HOLABYEd: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; '\$HOLA'\$'\$BYE'd</pre>				OK	OK	<pre>bash: HOLABYEd: command not found \$&gt;</pre>	127		
<pre>\$&gt; export HOLA=-n \$&gt; "echo \$HOLA" hey</pre>				OK	OK	<pre>bash: echo -n: command not found \$&gt;</pre>	127		
<pre>\$&gt; export A=1 B=2 C=3 D=4 E=5 F=6 G=7 H=8 \$&gt; echo "\$A'\$B' '\$C'\$D'\$E' '\$F' '\$G'\$H"</pre>				OK	OK	<pre>1'2\$C"\$D5"\$F'7'8 \$&gt;</pre>	0 0		

<pre>\$&gt; export HOLA=bonjour \$&gt; env \$&gt; unset HOLA \$&gt; env</pre>				OK	OK	<pre>... PATH=... \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; export HOLA=bonjour \$&gt; env \$&gt; unset HOLA \$&gt; unset HOLA \$&gt; env</pre>				OK	OK	<pre>... PATH=... \$&gt;</pre>	0 0 0 0 0		
<pre>\$&gt; unset PATH \$&gt; echo \$PATH</pre>				OK	OK	<pre>\$&gt;</pre>	0 0		
<pre>\$&gt; unset PATH \$&gt; ls</pre>				OK	OK	<pre>bash: ls: No such file or directory</pre>	127		
<pre>\$&gt; unset ""</pre>				OK	OK	<pre>bash: unset: `': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; unset INEXISTANT</pre>				OK	OK	<pre>\$&gt;</pre>	0		Cas où la variable n'existe pas
<pre>\$&gt; unset PWD \$&gt; env   grep PWD \$&gt; pwd</pre>				OK	OK	<pre>\$&gt; \$&gt; /home/vietdu91/42_works/minishell \$&gt;</pre>	0 0 0		
<pre>\$&gt; pwd \$&gt; unset PWD \$&gt; env   grep PWD \$&gt; cd \$PWD \$&gt; pwd</pre>				OK	OK	<pre>/home/vietdu91/42_works/minishell \$&gt; \$&gt; \$&gt; /home/vietdu91</pre>	0 0 0 0 0		
<pre>\$&gt; unset OLDPWD \$&gt; env   grep OLDPWD</pre>				OK	OK	<pre>\$&gt; \$&gt;</pre>	0 0		
<pre>\$&gt; unset 9HOLA</pre>				OK	OK	<pre>bash: unset: `9HOLA': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; unset HOLA9</pre>				OK	OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; unset HOL?A</pre>				OK	OK	<pre>bash: unset: `HOL?A': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; unset HOLA HOL?A</pre>				OK	OK	<pre>bash: unset: `HOL?A': not a valid identifier \$&gt;</pre>	1		Il unset bien HOLA
<pre>\$&gt; unset HOL?A HOLA</pre>				OK	OK	<pre>bash: unset: `HOL?A': not a valid identifier \$&gt;</pre>	1		Il unset bien HOLA
<pre>\$&gt; unset HOL?A HOL.A</pre>				OK	OK	<pre>bash: unset: `HOL?A': not a valid identifier bash: unset: `HOL.A': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; unset HOLA=</pre>				OK	OK	<pre>bash: unset: `HOLA=': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; unset HOL\\A</pre>	NON GERE			OK	OK	<pre>bash: unset: `HOL\\A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL;A</pre>	NON GERE			OK	OK	<pre>bash: A: command not found</pre>	127		
<pre>\$&gt; unset HOL.A</pre>				OK	OK	<pre>bash: unset: `HOL.A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL+A</pre>				OK	OK	<pre>bash: unset: `HOL+A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL=A</pre>				OK	OK	<pre>bash: unset: `HOL=A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL{A</pre>				OK	OK	<pre>bash: unset: `HOL{A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL}A</pre>				OK	OK	<pre>bash: unset: `HOL}A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL-A</pre>				OK	OK	<pre>bash: unset: `HOL-A': not a valid identifier</pre>	1		
<pre>\$&gt; unset -HOLA</pre>				OK	OK	<pre>bash: unset: -H: invalid option</pre>	2		
<pre>\$&gt; unset _HOLA</pre>				OK	OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; unset HOL_A</pre>				OK	OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; unset HOLA_</pre>				OK	OK	<pre>\$&gt;</pre>	0		
<pre>\$&gt; unset HOL*A</pre>				OK	OK	<pre>bash: unset: `HOL*A': not a valid identifier</pre>	1		
<pre>\$&gt; unset HOL#A</pre>				OK	OK	<pre>bash: unset: `HOL#A': not a valid identifier</pre>	1		
<pre>\$&gt; unset \$HOLA</pre>				OK	OK	<pre>\$&gt;</pre>	0		Il va unset le contenu de \$HOLA, c'est-a-dire la variable bonjour
<pre>\$&gt; unset \$PWD</pre>				OK	OK	<pre>bash: unset: `/home/vietdu91/42_works/minishell': not a valid identifier</pre>	1		

	\$> unset HOL@			OK	OK	bash: unset: `HOL@': not a valid identifier	1	
	\$> unset HOL!A			OK	OK	bash: !A: event not found	1	
	\$> unset HOL^A			OK	OK	bash: unset: `HOL^A': not a valid identifier	1	
	\$> unset HOL\$?A			OK	OK	\$>	0	Unset prend en compte le contenu de \$?, c'est-a-dire qu'il lit HOL0A ou HOL127A
	\$> unset HOL\~A			OK	OK	bash: unset: `HOL~': not a valid identifier	1	
	\$> unset "" HOLA			OK	OK	bash: unset: `': not a valid identifier	1	HOLA sera quand meme unset !!
	\$> env   grep HOLA					\$>	0	
	\$>					\$>	0	
	\$> unset PATH			OK	OK	\$>	0	
	\$> echo \$PATH						0	
	\$> unset PATH			OK	OK	bash: cat: No such file or directory	0	
	\$> cat Makefile					\$>	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	
FICHIERS BINAIRES 0 1	\$> /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					bash: /bin/cd: No such file or directory	127	
HISTORIQUE 	\$> history	NON GERE		OK	OK	... 558 echo \$? 559 clear 560 history \$>	0	Affiche les 500 dernières commandes
	\$> [touche du haut]			OK	OK	\$> clear	/	Defile l'historique
	\$> pwd			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	

CD  PWD	\$> 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 <a href="#">/home/vietdu91</a>
	\$> cd .			OK	OK	\$>	0	L'user reste dans le même dossier <a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd ./			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd ../../.			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd ../.././			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd ..			OK	OK	\$>	0	On se retrouve au dossier précédent <a href="#">/home/vietdu91/42_works</a>
	\$> cd ../			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works</a>
	\$> cd ../..			OK	OK	\$>	0	<a href="#">/home/vietdu91</a>
	\$> cd ../.			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works</a>
	\$> cd ../.././..			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works</a>
	\$> cd srcs			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell/srcs</a>
	\$> cd srcs objs			OK	OK	bash: cd: too many arguments \$>	1	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd 'srcs'			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell/srcs</a>
	\$> cd "srcs"			OK	OK	\$>	0	<a href="#">/home/vietdu91/42_works/minishell/srcs</a>
	\$> cd '/etc'			OK	OK	\$>	0	Marche aussi avec les double quotes <a href="#">/</a>
	\$> cd /e'tc'			OK	OK	\$>	0	<a href="#">/etc</a>
	\$> cd /e"tc"			OK	OK	\$>	0	<a href="#">/etc</a>
	\$> 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	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd ../../../../..			OK	OK	\$>	0	<a href="#">/</a>
	\$> cd ../../../.././bin/ls			OK	OK	bash: cd: ../../../.././bin/ls: No such file or directory \$>	1	<a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd /			OK	OK	\$>	0	<a href="#">/</a>
	\$> cd '/'			OK	OK	\$>	0	Marche aussi avec les double quotes <a href="#">/</a>
	\$> cd //	NON GERE		OK	OK	//	0	<a href="#">/</a>
	\$> pwd					\$>		
	\$> cd '//'	NON GERE		OK	OK	//	0	Marche aussi avec les double quotes <a href="#">/</a>
	\$> pwd					\$>		
	\$> cd ///			OK	OK	/	0	<a href="#">/</a>
	\$> pwd					\$>		
	\$> cd //////////			OK	OK	/	0	<a href="#">/</a>
	\$> pwd					\$>		
	\$> cd '/////////'			OK	OK	/	0	Marche aussi avec les double quotes <a href="#">/</a>
	\$> pwd					\$>		
	\$> cd /minishell			OK	OK	bash: cd: /minishell: No such file or directory \$>	1	
	\$> cd /			OK	OK	\$>	0	<a href="#">/</a>
	\$> cd ..			OK	OK	\$>		
	\$> 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 <a href="#">/home/vietdu91/42_works/minishell</a>
	\$> cd --	NON GERE		OK	OK	\$>	0	Retourne à \$HOME \$OLD_PWD et \$PWD sont inversés <a href="#">/home/vietdu91</a>

\$> 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 \$> cd \$HOME				OK	OK	bash: cd: HOME not set \$>	1		
\$> unset HOME \$> export HOME= \$> cd				OK	OK	\$>	0		/home/vietdu91/42_works/minishell
\$> unset HOME \$> export HOME \$> cd				OK	OK	bash: cd: HOME not set \$>	1		
\$> cd minishell Docs crashtest.c				OK	OK	bash: cd: too many arguments \$>	1		
\$> cd /   echo \$? \$> pwd				OK	OK	0 \$>	0 0		/home/vietdu91/42_works/minishell
\$> cd ~				OK	OK	\$>	0		/home/vietdu91
\$> cd ~/	NON GERE			OK	OK	\$>	0		/home/vietdu91
\$> cd ~/   echo \$? \$> pwd	NON GERE			OK	OK	0 \$>	0		/home/vietdu91/42_works/minishell
\$> cd *	NON GERE			OK	OK	\$>	0		[BONUS] S'il n'y a qu'un seul <u>dossier</u> et rien d'autre
\$> cd *	NON GERE			OK	OK	bash: cd: minishell: No such file or directory \$>	1		[BONUS] S'il n'y a qu'un seul <u>fichier</u> et rien d'autre
\$> cd *	NON GERE			OK	OK	bash: cd: too many arguments \$>	1		[BONUS] S'il y au moins deux éléments
\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r ../../a \$> cd ..				OK	OK	chdir: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory \$>	1		
\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r ../../a \$> pwd				OK	OK	pwd: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory \$>	1		
\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r ../../a \$> echo \$PWD \$> echo \$OLDPWD				OK	OK	/home/vietdu91/42_works/minishell/a/b \$> /home/vietdu91/42_works/minishell \$>	0 0		
\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r ../../a \$> cd \$> echo \$PWD \$> echo \$OLDPWD				OK	OK	/home \$> /home/vietdu91/42_works/minishell/a/b \$>	0 0		
\$> mkdir a \$> cd a \$> rm -r ../a \$> echo \$PWD \$> echo \$OLDPWD				OK	OK	../../../../home/vietdu91/42_works/minishell/a \$> ../../../../home/vietdu91/42_works/minishell \$>	0 0		
\$> export CDPATH= \$> cd \$HOME/..	NON GERE			OK	OK	\$>	0 0		A gerer ? /home



	<pre>\$&gt; export CDPATH=/ \$&gt; cd /home/vietdu91</pre>	NON GERE			OK	OK	<pre>/home/vietdu91 \$&gt;</pre>	0 0		A gerer ? /home/vietdu91
	<pre>\$&gt; export CDPATH=./ \$&gt; cd .</pre>	NON GERE			OK	OK	<pre>\$&gt;</pre>	0 0		A gerer ? /home/vietdu91/42_works/minishell
	<pre>\$&gt; export CDPATH=./ \$&gt; cd ..</pre>	NON GERE			OK	OK	<pre>\$&gt;</pre>	0 0		A gerer ? /home/vietdu91/42_works
<div><div></div><div>BÂTARDS</div></div>	<pre>\$&gt; chmod 000 minishell \$&gt; ./minishell</pre>				OK	OK	<pre>bash: ./minishell : Permission denied \$&gt;</pre>	126		
	<pre>\$&gt; ls hola</pre>				OK	OK	<pre>ls: cannot access 'hola': No such file or directory \$&gt;</pre>	2		
	<pre>\$&gt; ./Makefile</pre>				OK	OK	<pre>bash: ./Makefile: Permission denied \$&gt;</pre>	126		
	<pre>\$&gt; ./minishell</pre>				OK	OK	<pre>\$&gt;</pre>	0		Relance le Minishell Il faut donc 2 fois exit pour exit
	<pre>\$&gt; env   grep SHLVL \$&gt; ./minishell \$&gt; env   grep SHLVL \$&gt; exit \$&gt; env   grep SHLVL</pre>	NON GERE			OK	OK	<pre>\$&gt;</pre>	0 0 0 0 0		
	<pre>\$&gt; touch hola \$&gt; ./hola</pre>				OK	OK	<pre>bash: ./hola: Permission denied \$&gt;</pre>	126		
	<pre>\$&gt; env "wc" -l</pre>				OK	OK	<pre>66 \$&gt;</pre>	0		
	<pre>\$&gt; env "wc "-l</pre>				OK	OK	<pre>bash: wc -l: command not found \$&gt;</pre>	127		
	<pre>\$&gt; expr 1 + 1</pre>				OK	OK	<pre>2 \$&gt;</pre>	0		
	<pre>\$&gt; expr \$? + \$?</pre>				OK	OK	<pre>0 \$&gt;</pre>	0		Fait la somme de la valeur de \$?
	<pre>\$&gt; env -i ./minishell \$&gt; env</pre>				OK	OK	<pre>PWD=/mnt/nfs/homes/emtran/Desktop/42_minishell SHLVL=1 _=/usr/bin/env \$&gt;</pre>	0		
	<pre>\$&gt; env -i ./minishell \$&gt; export</pre>				OK	OK	<pre>export OLDPWD export PWD="/mnt/nfs/homes/emtran/Desktop/42_minishell" export SHLVL="1" \$&gt;</pre>	0		
	<pre>\$&gt; env -i ./minishell \$&gt; cd</pre>				OK	OK	<pre>bash: cd: HOME not set \$&gt;</pre>	1		
	<pre>\$&gt; env -i ./minishell \$&gt; cd ~</pre>	NON GERE			OK	OK	<pre>\$&gt;</pre>	0		/home/vietdu91
	<pre>\$&gt; exit</pre>				OK	OK	<pre>exit ~/minishell \$&gt;</pre>	0		
	<pre>\$&gt; exit exit</pre>				OK	OK	<pre>exit bash: exit: exit: numeric argument required ~/minishell \$&gt;</pre>	2		
	<pre>\$&gt; exit hola</pre>				OK	OK	<pre>exit bash: exit: hola: numeric argument required ~/minishell \$&gt;</pre>	2		
	<pre>\$&gt; exit hola que tal</pre>				OK	OK	<pre>exit bash: exit: hola: numeric argument required ~/minishell \$&gt;</pre>	2		
	<pre>\$&gt; exit 42</pre>				OK	OK	<pre>exit ~/minishell \$&gt;</pre>	42		
	<pre>\$&gt; exit 000042</pre>				OK	OK	<pre>exit ~/minishell \$&gt;</pre>	42		
	<pre>\$&gt; exit 666</pre>				OK	OK	<pre>exit ~/minishell \$&gt;</pre>	154		
	<pre>\$&gt; exit 666 666</pre>				OK	OK	<pre>exit bash: exit: too many arguments \$&gt;</pre>	1		

\$> exit -666 666				OK	OK	exit bash: exit: too many arguments \$>	1		
\$> exit hola 666				OK	OK	exit bash: exit: hola: numeric argument required ~/minishell \$>	2		
\$> exit 666 666 666 666				OK	OK	exit bash: exit: too many arguments \$>	1		
\$> exit 666 hola 666				OK	OK	exit bash: exit: too many arguments	1		
\$> exit hola 666 666				OK	OK	exit bash: exit: hola: numeric argument required ~/minishell \$>	2		
\$> exit 259				OK	OK	exit ~/minishell \$>	3		
\$> exit -4				OK	OK	exit ~/minishell \$>	252		
\$> exit -42				OK	OK	exit ~/minishell \$>	214		
\$> exit -0000042				OK	OK	exit ~/minishell \$>	214		
\$> exit -259				OK	OK	exit ~/minishell \$>	253		
\$> exit -666				OK	OK	exit ~/minishell \$>	102		
\$> exit +666				OK	OK	exit ~/minishell \$>	154		
\$> exit 0				OK	OK	exit ~/minishell \$>	0		
\$> exit +0				OK	OK	exit ~/minishell \$>	0		
\$> exit -0				OK	OK	exit ~/minishell \$>	0		
\$> exit +42				OK	OK	exit ~/minishell \$>	42		
\$> exit -69 -96				OK	OK	exit bash: exit: too many arguments \$>	1		
\$> exit --666				OK	OK	exit bash: exit: --666: numeric argument required ~/minishell \$>	2		
\$> exit ++++666				OK	OK	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2		
\$> exit ++++++0				OK	OK	exit bash: exit: ++++++0: numeric argument required ~/minishell \$>	2		
\$> exit -----0				OK	OK	exit bash: exit: -----0: numeric argument required ~/minishell \$>	2		
\$> exit "666"				OK	OK	exit ~/minishell \$>	154		
\$> exit '666'				OK	OK	exit ~/minishell \$>	154		
\$> exit '-666'				OK	OK	exit ~/minishell \$>	102		Marche aussi pour les double quotes
\$> exit '+666'				OK	OK	exit ~/minishell \$>	154		Marche aussi pour les double quotes
\$> exit '----666'				OK	OK	exit bash: exit: ----666: numeric argument required ~/minishell \$>	2		Marche aussi pour les double quotes

	\$> exit '++++666'			OK	OK	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2	Marche aussi pour les double quotes
	\$> exit '6'66			OK	OK	exit ~/minishell \$>	154	Marche aussi pour les double quotes
	\$> exit '2'66'32'			OK	OK	exit ~/minishell \$>	8	Marche aussi pour les double quotes
	\$> exit "'666'"			OK	OK	exit bash: exit: '666': numeric argument required ~/minishell \$>	2	
	\$> exit '"666"'			OK	OK	exit bash: exit: "666": numeric argument required ~/minishell \$>	2	
	\$> exit '666'"666"666			OK	OK	exit ~/minishell \$>	170	
	\$> exit +'666'"666"666			OK	OK	exit ~/minishell \$>	170	
	\$> exit -'666'"666"666			OK	OK	exit ~/minishell \$>	86	
	\$> exit 9223372036854775807			OK	OK	exit ~/minishell \$>	255	
	\$> exit 9223372036854775808			OK	OK	exit bash: exit: 9223372036854775808: numeric argument required ~/minishell \$>	2	
	\$> exit -9223372036854775808			OK	OK	exit ~/minishell \$>	0	
	\$> exit -9223372036854775809			OK	OK	exit bash: exit: -9223372036854775809: numeric argument required ~/minishell \$>	2	
	\$> cat   cat   cat   ls			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c  \$>	0	Apparaît la ligne ls Et 3 retours à la ligne avec la touche Entrée
	\$> ls   exit			OK	OK	\$>	0	Ne fait rien  Chaque commande d'un pipeline est exécutée dans son propre sous-shell, qui est un processus distinct
	\$> ls   exit 42			OK	OK	\$>	42	Ne fait rien
	\$> exit   ls			OK	OK	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	
	\$> echo hola > bonjour \$> exit   cat -e bonjour			OK	OK	hola\$ \$>	0	
	\$> echo hola > bonjour \$> cat -e bonjour   exit			OK	OK	\$>	0	
	\$> echo   echo			OK	OK	\$>	0	
	\$> echo hola   echo que tal			OK	OK	que tal \$>	0	
	\$> pwd   echo hola			OK	OK	hola \$>	0	
	\$> env   echo hola			OK	OK	hola \$>	0	
	\$> echo oui   cat -e			OK	OK	oui\$ \$>	0	
	\$> echo oui   echo non   echo hola   grep oui			OK	OK	\$>	0	
	\$> echo oui   echo non   echo hola   grep non			OK	OK	\$>	0	



	\$> echo hola    cat	NON GERE		OK	OK	hola \$>	0	[BONUS]
	\$> echo hola     cat			OK	OK	bash: syntax error near unexpected token ` ' \$>	2	
	\$> ech o hola   cat			OK	OK	bash: ech: command not found bash: o: command not found \$>	127	
	\$> cat Makefile   cat -e   cat -e			OK	OK	[...] \$\$ .PHONY:    all clean fclean re\$\$ \$>	0	
	\$> cat Makefile   grep srcs   cat -e			OK	OK	objs/%.o:          srcs/%.c\$ @norminette srcs/*\$ \$>	0	
	\$> cat Makefile   grep srcs   grep srcs   cat -e			OK	OK	objs/%.o:          srcs/%.c\$ @norminette srcs/*\$ \$>	0	
	\$> cat Makefile   grep pr   head -n 5   cd file_not_exist			OK	OK	bash: cd: file_not_exist: No such file or directory \$>	1	
	\$> cat Makefile   grep pr   head -n 5   hello			OK	OK	bash: hello: command not found \$>	127	
	\$> export HOLA=bonjour   cat -e   cat -e			OK	OK	\$>	0	
	\$> unset HOLA   cat -e			OK	OK	\$>	0	
	\$> export HOLA   echo hola \$> env   grep PROUT			OK	OK	hola \$> \$>	0	
	\$> export   echo hola			OK	OK	hola \$>	0	
	\$> sleep 3   sleep 3			OK	OK	\$>	0	Le prompt apparait au bout de 3 secondes
	\$> time sleep 3   sleep 3			OK	OK	real    0m3.002s user    0m0.003s sys     0m0.001s \$>	0	Le prompt apparait au bout de 3 secondes
	\$> sleep 3   exit			OK	OK	\$>	0	Le prompt apparait au bout de 3 secondes
	\$> exit   sleep 3			OK	OK	\$>	0	Le prompt apparait au bout de 3 secondes
	\$> echo hola > a \$> >>b echo que tal \$> cat a   <b cat   cat > c   cat			OK	OK	\$>	0	
	\$> pwd && ls	NON GERE		OK	OK	/home/vietdu91/42_works/minishell crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0	[BONUS]
	\$> pwd    ls	NON GERE		OK	OK	/home/vietdu91/42_works/minishell \$>	0	[BONUS]
	\$> echo hola    echo bonjour	NON GERE		OK	OK	hola \$>	0	[BONUS]
	\$> echo hola && echo bonjour	NON GERE		OK	OK	hola bonjour \$>	0	[BONUS]
	\$> echo bonjour    echo hola	NON GERE		OK	OK	bonjour \$>	0	[BONUS]
	\$> echo bonjour && echo hola	NON GERE		OK	OK	bonjour hola \$>	0	[BONUS]
	\$> echo -n bonjour && echo -n hola	NON GERE		OK	OK	bonjourhola\$>	0	[BONUS]
	\$> pwd && ls && echo hola	NON GERE		OK	OK	/home/vietdu91/42_works/minishell crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c hola \$>	0	[BONUS]



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

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



<pre>\$&gt; (ls &gt; Docs/hey &amp;&amp; pwd) &gt; hola \$&gt; cat hola \$&gt; cat Docs/hey</pre>	NON GERE			OK	OK	<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			OK	OK	<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			OK	OK	<pre>/home/vietdu91/ /home/vietdu91/ \$&gt;</pre>	0		[BONUS]
<pre>\$&gt; (cd ../../ &amp;&amp; pwd) &amp;&amp; pwd</pre>	NON GERE			OK	OK	<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			OK	OK	<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			OK	OK	<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	OK	<pre>hola \$&gt;</pre>	0 0		
<pre>\$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat bonjour</pre>				OK	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	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	OK	<pre>que tal \$&gt;</pre>	0 0 0 0		
<pre>\$&gt; echo hola que tal &gt; bonjour \$&gt; cat bonjour</pre>				OK	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	OK	<pre>hola que tal\$ \$&gt;</pre>	0 0		
<pre>\$&gt; export HOLA=hey \$&gt; echo bonjour &gt; \$HOLA \$&gt; echo \$HOLA</pre>				OK	OK	<pre>\$&gt; hola \$&gt;</pre>	0 0 0		Ne fait rien
<pre>\$&gt; whereis grep &gt; Docs/bonjour \$&gt; cat Docs/bonjour</pre>				OK	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	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	OK	<pre>/home/vietdu91/42_works/minishell \$&gt;</pre>	0 0		

	\$> pwd >                    bonjour \$> cat bonjour				OK	OK	/home/vietdu91/42_works/minishell \$>	0 0		
	\$> echo hola > > bonjour				OK	OK	bash: syntax error near unexpected token `>' \$>	2		
	\$> echo hola < < bonjour				OK	OK	bash: syntax error near unexpected token `<' \$>	2		
	\$> echo hola >>> bonjour				OK	OK	bash: syntax error near unexpected token `>' \$>	2		
	\$> > bonjour echo hola \$> cat bonjour				OK	OK	hola \$>	0 0		
	\$> > bonjour   echo hola \$> cat bonjour				OK	OK	hola \$> \$>	0 0		
	\$> prout hola > bonjour \$> ls				OK	OK	bash: prout: command not found \$> bonjour [...]	127 0		Erreur mais file "bonjour" cree tout de meme
	\$> echo hola > hello >> hello >> hello \$> ls \$> cat hello				OK	OK	\$> hello \$> hola \$>	0 0 0		
	\$> echo hola > hello >> hello >> hello \$> echo hola >> hello \$> cat < hello				OK	OK	hola hola \$>	0 0 0		
	\$> echo hola > hello >> hello >> hello \$> echo hola >> hello \$> echo hola > hello >> hello >> hello \$> cat < hello				OK	OK	hola \$>	0 0 0 0		
	\$> echo hola >> hello >> hello > hello \$> echo hola >> hello \$> cat < hello				OK	OK	hola hola \$>	0 0 0		
	\$> echo hola >> hello >> hello > hello \$> echo hola >> hello \$> echo hola >> hello >> hello > hello \$> cat < hello				OK	OK	hola \$>	0 0 0 0		
	\$> echo hola > hello \$> echo hola >> hello >> hello >> hello \$> echo hola >> hello \$> cat < hello				OK	OK	hola hola hola \$>	0 0 0 0		
	\$> echo hola > hello \$> echo hey > bonjour \$> echo <bonjour <hello				OK	OK	\$>	0 0 0		
	\$> echo hola > hello \$> echo hey > bonjour \$> echo <hello <bonjour				OK	OK	\$>	0 0 0		
	\$> echo hola > bonjour \$> echo hey > hello \$> rm bonjour hello \$> echo hola > bonjour > hello > bonjour \$> cat bonjour \$> cat hello				OK	OK	hola \$> \$>	0 0 0 0 0 0		
	\$> echo hola > bonjour \$> echo hey > hello \$> echo hola > bonjour > hello > bonjour \$> cat bonjour \$> cat hello				OK	OK	hola \$> \$>	0 0 0 0		
	\$> echo hola > bonjour \$> echo hey > hello \$> rm bonjour hello \$> echo hola > bonjour >> hello > bonjour \$> cat bonjour \$> cat hello				OK	OK	hola \$> \$>	0 0 0 0 0 0		

<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>				OK	OK	<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>				OK	OK	<pre>hola \$&gt; \$&gt;</pre>	0 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>				OK	OK	<pre>hola \$&gt; \$&gt;</pre>	0 0 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>				OK	OK	<pre>hola \$&gt; \$&gt;</pre>	0 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>				OK	OK	<pre>hola \$&gt; \$&gt;</pre>	0 0 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>				OK	OK	<pre>hola \$&gt; \$&gt;</pre>	0 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>				OK	OK	<pre>hola hola \$&gt; hey \$&gt;</pre>	0 0 0 0 0 0 0		
<pre>\$&gt; &gt; bonjour echo hola bonjour \$&gt; cat bonjour</pre>				OK	OK	<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>				OK	OK	<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>				OK	OK	<pre>hola1 hola2 [...] \$&gt; \$&gt; hello 2 \$&gt;</pre>	0 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>				OK	OK	<pre>hola1 hola2 [...] \$&gt; bonjour \$&gt; 2 \$&gt;</pre>	0 0 0 0 0 0 0		
<pre>\$&gt; &gt; pwd \$&gt; ls</pre>				OK	OK	<pre>pwd \$&gt;</pre>	0 0		
<pre>\$&gt; &lt; pwd</pre>				OK	OK	<pre>bash: pwd: No such file or directory \$&gt;</pre>	1		
<pre>\$&gt; &lt; Makefile .</pre>	NON GERE			OK	OK	<pre>.: not enough arguments \$&gt;</pre>	1		Voir pourquoi
<pre>\$&gt; cat &lt;pwd</pre>				OK	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; rm srcs/bonjour srcs/hello \$&gt; &gt;srcs/bonjour &gt;srcs/hello &lt;prout \$&gt; ls srcs \$&gt; cat srcs/bonjour srcs/hello</pre>				OK	OK	<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; 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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<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>				OK	OK	<pre>hola \$&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>				OK	OK	<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>				OK	OK	<pre>bash: prout: No such file or directory \$&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>				OK	OK	<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 hello \$&gt; cat bonjour</pre>				OK	OK	<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>				OK	OK	<pre>0 0 0 \$&gt;</pre>	0 0 0		
<pre>\$&gt; rm -f bonjour \$&gt; rm bonjour &gt; bonjour \$&gt; ls -l bonjour</pre>				OK	OK	<pre>ls: cannot access 'bonjour': No such file or directory</pre>	0 0 2		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA \$&gt; ls</pre>				OK	OK	<pre>'bonjour hello' [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;"\$HOLA" \$&gt; ls</pre>				OK	OK	<pre>'bonjour hello' [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$"HOLA" \$&gt; ls</pre>				OK	OK	<pre>HOLA [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA&gt;hey \$&gt; ls</pre>				OK	OK	<pre>'bonjour hello' hey [...] \$&gt;</pre>	0 0 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;hey&gt;\$HOLA \$&gt; ls</pre>				OK	OK	<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>				OK	OK	<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>				OK	OK	<pre>\$&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>				OK	OK	<pre>Prout \$&gt;</pre>	0		
<pre>\$&gt; &lt;a cat &lt;b &lt;c \$&gt; cat a \$&gt; cat b \$&gt; cat c</pre>				OK	OK	<pre>\$&gt; \$&gt; \$&gt; Amour Tu es Horrible 0123456789 Prout \$&gt;</pre>	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>				OK	OK	<pre>\$&gt; \$&gt; \$&gt; Prout \$&gt; crashtest.c Docs [...] test.c \$&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>				OK	OK	<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>				OK	OK	<pre>\$&gt; \$&gt; \$&gt; hola \$&gt;</pre>	0		
<pre>\$&gt; mkdir dir \$&gt; ls -la &gt; dir/bonjour \$&gt; cat dir/bonjour</pre>				OK	OK	<pre>total 16 drwx----- 2 vietdu91 [...] . drwxrwxewt 51 root    [...] .. [...] \$&gt;</pre>	0		
<pre>\$&gt; &lt;a \$&gt; cat a</pre>				OK	OK	<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>				OK	OK	<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>				OK	OK	<pre>Amour Tu Es Horrible \$&gt; [...] a b hey d</pre>	0 0 0		
<pre>\$&gt; cat &lt;&lt; hola</pre>				OK	OK	<pre>&gt; \$HOME &gt; hola /home/vietdu91 \$&gt;</pre>	0		
<pre>\$&gt; cat &lt;&lt; 'hola'</pre>				OK	OK	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0		

<div>HEREDOC</div>	<pre>\$&gt; cat &lt;&lt; "hola"</pre>			OK	OK	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0	
	<pre>\$&gt; cat &lt;&lt; ho"la"</pre>			OK	OK	<pre>&gt; \$HOME &gt; hola \$HOME \$&gt;</pre>	0	
	<pre>\$&gt; cat &lt;&lt; \$HOME</pre>			OK	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	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	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	OK	<pre>&gt; chola &gt; holache &gt;   hola &gt; "hola" &gt; 'hola' &gt; hola \$&gt;</pre>	0	
	<pre>\$&gt; &lt;&lt;hola</pre>			OK	OK	<pre>&gt; &gt; &gt; hola \$&gt;</pre>	0	
	<pre>\$&gt; cat &lt;&lt;</pre>			OK	OK	<pre>bash: syntax error near unexpected token `newline' \$&gt;</pre>	2	
	<pre>\$&gt; cat &lt;&lt; prout &lt;&lt; lol &lt;&lt; koala</pre>			OK	OK	<pre>&gt; prout &gt; lol &gt; koala \$&gt;</pre>	0	
	<pre>\$&gt; prout &lt;&lt; lol &lt;&lt; cat &lt;&lt; koala</pre>			OK	OK	<pre>&gt; prout &gt; lol &gt; cat &gt; koala bash: prout: command not found</pre>	127	
	<pre>\$&gt; &lt;&lt; \$hola</pre>			OK	OK	<pre>&gt; \$hola \$&gt;</pre>	0	
	<pre>\$&gt; &lt;&lt; \$"hola"\$"b"</pre>			OK	OK	<pre>&gt; holab \$&gt;</pre>	0	
	<pre>\$&gt; &lt;&lt; \$"\$hola"\$"\$b"</pre>			OK	OK	<pre>&gt; \$hola\$\$b \$&gt;</pre>	0	
	<pre>\$&gt; &lt;&lt; ho\$la\$"\$a"\$"\$b"</pre>			OK	OK	<pre>&gt; hola\$a\$\$b &gt; ho\$la\$a\$\$nb \$&gt;</pre>	0	
	<pre>\$&gt; echo hola &lt;&lt;&lt; bonjour</pre>			OK	OK	<pre>hola \$&gt;</pre>	0	
	<pre>\$&gt; echo hola &lt;&lt;&lt;&lt; bonjour</pre>			OK	OK	<pre>bash: syntax error near unexpected token `&lt;' \$&gt;</pre>	2	



	\$> echo hola <<<< bonjour				OK	OK	bash: syntax error near unexpected token `<<' \$>	2	
WILDCARD ⭐	\$> ls *	NON GERE			OK	OK	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			OK	OK	crashtest.c test.c \$>	0	[BONUS]
	\$> ls *.hola	NON GERE			OK	OK	ls: cannot access '*.hola' : No such file or directory \$>	2	[BONUS]
	\$> cat M*le	NON GERE			OK	OK	[affiche 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	[affiche le Makefile] \$>	0	[BONUS]