## **MINISHELL**

## **ATTENTION**

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

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

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

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

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

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

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

## Gestion des comportements

	A faire	A tester	Leaks ?	Fonctionne		g II ( 1 +2)	Notes	
test.c	Х	х	ок ко	ок ко	Comportement attendu	Sortie d'exit (echo \$?)		
\$> \n (touche entrée)			ОК	ОК	\$>	Exit de l'ancienne commande		
\$> [que des espaces]			ок	ок		0		
<pre>\$&gt; [que des tabulations]</pre>			ОК	ок		0	Crl + V + Tab	
\$> :			ОК	ок		0	Marche aussi pour "#"	
\$> !			ОК	ок		1		
\$> >			ОК	ОК	bash: syntax error near unexpected token `newline'	2		
\$> <			ОК	ОК	\$>	2		
\$> >>			ОК	ОК		2		
\$> <<			ОК	ОК		2		
\$> <>			ОК	ОК		2		
\$> >>>>				ок	bash: syntax error near unexpected token `>>' \$>	2		
\$> >>>>>>>>			ок	ок	bash: syntax error near unexpected token `>>' \$>	2		
\$> <<<<			ок	ок	bash: syntax error near unexpected token `<<' \$>	2		
\$> ececcicceccicc			ок	ОК	bash: syntax error near unexpected token `<<' \$>	2		
\$>>>>			ок	ОК	bash: syntax error near unexpected token `>' \$>	2	Marche aussi pour <	
\$> >> >> >>				ок	bash: syntax error near unexpected token `>>' \$>	2	Marche aussi pour <	
\$> >>>> >> >>			ок	ок	bash: syntax error near unexpected token `>>' \$>	2	Marche aussi pour <	
\$> /			ок	ок	bash: /: Is a directory \$>	126		
\$> //			ок	ок	bash: //: Is a directory \$>	126		

	\$> /.			ОК	ОК	bash: /.: Is a directory \$>	126		
	\$> /.///				ок	bash: /.////	126		
	\$>				ОК	bash: //////: Is a directory	126		
	\$> \\	NON GERE			ОК	bash: \: command not found	127		
	\$> \\\\	NON GERE			ОК	bash: \\: command not found	127		
	\$> \\\\\\	NON GERE			ОК	bash: \\\\: command not found	127		
	\$> -				ок	bash: -: command not found \$>	127	Marche aussi pour "_@°]=+£\$?"	
	\$>				ОК	bash: syntax error near unexpected token ` '	2	Marche aussi pour les metacharacters "&();"	
	\$>   hola				ОК	bash: syntax error near unexpected token ` '	2		
(XE)	\$>				ОК	bash: syntax error near unexpected token ` '	2		
E (SYNTA	\$>				ОК	bash: syntax error near unexpected token `  '	2		
LAVOLE	\$>				ОК	bash: syntax error near unexpected token `  '	2		
TERES A	\$>				ОК	bash: syntax error near unexpected token `  '	2		
CARAC	\$> >> ><				ОК	bash: syntax error near unexpected token ` '	2		
	\$> &&				ОК	bash: syntax error near unexpected token `&&'	2		
	\$> &&&&&				ОК	bash: syntax error near unexpected token `&&' \$>	2		
	\$> &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&				ОК	bash: syntax error near unexpected token `&&' \$>	2		
	\$> ;;	NON GERE			ОК	bash: syntax error near unexpected token `;;' \$>	2		
	\$> ;;;;;	NON GERE			ОК	bash: syntax error near unexpected token `;;' \$>	2		
	\$> ;;;;;;;;;;;	NON GERE			ок	bash: syntax error near unexpected token `;;' \$>	2		
	\$> ()	NON GERE			ОК	bash: syntax error near unexpected token `)' \$>	2		
	\$> ( ( ) )	NON GERE			ОК	<pre>bash: syntax error near unexpected token `)' \$&gt;</pre>	2		
	\$> ( ( ( ( ) ) ) )	NON GERE			ОК	bash: syntax error near unexpected token `)' \$>	2		
	\$> ""	NON GERE			ок	bash: : command not found \$>	127		
	\$> "hola"				ок	bash: hola: command not found \$>	127		
	\$> 'hola'				ок	bash: hola: command not found \$>	127		
	\$> ' '	NON GERE			ОК	bash: : command not found \$>	127		
	\$> *	NON GERE			ОК	bash: crashtest.c: command not found	127	[BONUS]	
	\$> */*	NON GERE			ОК	bash: Docs/attest.cation: command not found	127	[BONUS]	
	\$> */*	NON GERE			ОК	bash: */*: No such file or directory	127	[BONUS] S'il n'y aucun dossier à l'intérieur	
	\$> .	NON GERE			ОК	bash: .: filename argument required	2		
						\$>			

\$>	NON GERE	OK	01/	bash:: command not found	127		
		ОК	ОК	\$>			
\$> ~	NON GERE	ОК	ОК	<pre>bash: /home/vietdu91: Is a directory \$&gt;</pre>	126		
\$> ABC=hola	NON GERE	ОК	ок	\$>	0	/!\ A gerer comme une erreur	
\$> 4ABC=hola	NON GERE	ок	ок	bash: 4ABC=hola: command not found	127		
hola		ОК	ок	bash: hola: command not found	127		
\$> hola que tal		ОК	ок	bash: hola: command not found \$>	127		
\$> Makefile		ОК	ок	bash: Makefile: command not found \$>	127		
\$> echo		ОК	ок	\$>	0		
\$> echo -n		ОК	ок	\$>	0		
\$> echo Hola		ок	ок	Hola \$>	0		
\$> echoHola		ок	ок	bash: echoHola: command not found \$>	127		
\$> echo-nHola		ок	ок	bash: echo-nHola: command not found \$>	127		
\$> echo -n Hola		ОК	ОК	Hola\$>	0		
\$> echo "-n" Hola		ОК	ок	Hola\$>	Θ		
\$> echo -nHola		ОК	ок	-nHola \$>	0		
\$> echo Hola -n		ок	ок	Hola -n \$>	Θ		
\$> echo Hola Que Tal		ОК	ок	Hola Que Tal \$>	Θ		
\$> echo Hola		ОК	ок	Hola \$>	Θ		
\$> echo Hola Que Tal		ОК	ок	Hola Que Tal \$>	Θ		
\$> echo \n hola		ок	ок	n hola \$>	0	A gerer ?	
\$> echo " "   cat -e		ОК	ок	\$	0		
\$> echo   cat -e		ок	ок	\$ \$>	0		
\$> ""''echo hola""''' que""'' tal""''		ОК	ок	hola que tal \$>	0		
\$> echo -n -n		ОК	ок	\$>	0		
\$> echo -n -n Hola Que		ОК	ок	Hola Que\$>	0		
\$> echo -p		ОК	ок	-p \$>	Θ		
\$> echo -nnnnn		ОК	ок	\$>	0		
\$> echo -n -nnn -nnnn		ОК	ок	\$>	0		
\$> echo -n-nnn -nnnn		ОК	ок	-n-nnn -nnnn \$>	0		
\$> echo -n -nnn hola -nnnn		ОК	ОК	hola -nnnn\$>	0		
\$> echo -n -nnn-nnnn		ОК	ок	-nnn-nnnn\$>	Θ		
\$> echon		ок	ок	\$>	0		
\$> echo -nnnn		ОК	ок	n\$>	0		
\$> echo -nnnnnnn		ОК	ОК	nnnnnn\$>	0		
\$> echo -nnnnnnn		ОК	ОК	nnnn\$>	0		
\$> echo \$		ОК	ок	\$ \$>	0		
\$> echo \$?		ок	ок	0 \$>		Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan	
\$> echo \$?\$		ок	ок	0\$ \$>	0		
\$> echo \$?   echo \$?   echo \$?		ок	ок	0 \$>	0		
\$> echo \$:\$=   cat -e		ок	ок	\$:\$=\$ \$>	0		

\$> echo "\$"""	ОК	ОК	\$ \$>	9	
\$> echo '\$'''	ок	ок	\$ \$>	9	
\$> echo \$"HOME"	ок	ок	HOME \$>	9	
\$> echo \$''HOME	ок	ок	HOME \$>	9	
\$> echo \$""HOME	ок	ок		9	
\$> echo "\$HO"ME	ОК	ОК		9	
\$> echo '\$HO'ME	ОК	ОК		9	
\$> echo "\$HO""ME"	ОК	ОК		9	
\$> echo '\$HO''ME'	ОК	ОК		9	
\$> echo "'\$HO''ME'"	OK	OK	''ME'	Il y a bien 3 singles quotes avant le ME. Je	
	ОК	ОК	37	ne sais pas pourquoi Excel tronque à l'affichage le 3e quote	
\$> echo ""\$HOME	ОК	ок	\$>	9	
\$> echo "" \$HOME	ОК	ОК	/home/vietdu91 \$>		
\$> echo ''\$HOME	ОК	ок	\$>	9	
\$> echo '' \$HOME	ОК	ок	/home/vietdu91 \$>		
\$> echo \$"HO""ME"	ок	ок	HOME \$>	9	
\$> echo \$'H0''ME'	ОК	ОК	HOME \$>	9	
\$> echo \$'HOME'	ок	ОК	HOME \$>	9	
\$> echo "\$"HOME	ок	ок	SHOME 6	9	
\$> echo \$=HOME	ок	ок	\$=HOME 6 \$>	9	
\$> echo \$"HOLA"	ОК	ок	HOLA S>	9	
\$> echo \$'HOLA'	ОК	ок		9	
\$> echo \$DONTEXIST Hola	ОК	ОК		Le cas ou la variable n'existe pas et n'est pas importee	
\$> echo "hola"	ОК	ОК		9	
\$> echo 'hola'	ОК	ОК		9	
\$> echo ''hola''	ОК	ок	hola	9	
\$> echo ''h'o'la''	ОК	ок		9	
\$> echo "''h'o'la''"	ОК	ок		9	
\$> echo "'"h'o'la"'"	ок	ок		9	
\$> echo"'hola'"	ок	ОК		9	
\$> echo "'hola'"	ок	ОК		9	
\$> echo '"hola"'	ок	ок		9	
\$> echo '''ho"'''l"a'''	ОК	ОК	\$> ho"l"a	9	
\$> echo hola""""""""""""""""""""""""""""""""""""	OK OK	OK	\$>	9	
\$> echo hola"'''''			\$>	9	
\$> echo hola'''''''	OK	ОК	\$>	9	
\$> echo hola'************************************	ОК	ОК	\$>	9	
\$2 ecilo liota	ОК	ок	nota		

\$> e"cho hola"	ок ок	bash: echo hola: command not found \$>	127	
\$> e'cho hola'	ок ок	bash: echo hola: command not found \$>	127	
\$> echo "hola "   cat -e	ок ок	hola \$ \$>	0	
\$> echo ""hola	ок ок	hola \$>	0	
\$> echo "" hola	ок ок	hola \$>	0	
\$> echo "" hola	ок ок	hola S>	0	
\$> echo ""hola	ок ок	hola \$>	0	
\$> echo "" hola	ок ок	hola \$>	0	
\$> echo hola""bonjour	ок ок	holabonjour \$>	0	
\$> "e"'c'ho 'b'"o"nj"o"'u'r	ок ок	bonjour S>	0	
\$> ""e"'c'ho 'b'"o"nj"o"'u'r"	ок ок	bash: e'c'ho 'b'onjo'u'r: command not found	127	
\$> echo "\$DONTEXIST"Makefile	ок ок	\$> Makefile	0	
\$> echo "\$DONTEXIST""Makefile"	ок ок	\$> Makefile \$>	0	
\$> echo "\$DONTEXIST" "Makefile"	ок ок	Makefile	0	Attention a l'espace avant
\$> \$?	ок ок	\$> bash: 0: command not found	127	Renvoie le code de retour du dernier pipeline exécuté à l'avant-plan
\$> \$7\$?	ок ок	\$> bash: 00: command not found \$>	127	execute a t avant-plan
\$> ?\$HOME	ок ок	bash: ?/home/vietdu91: command not found	127	
\$> \$	ок ок	\$> bash: \$: command not found	127	
\$> \$HOME	ок ок	\$> bash: /home/vietdu91: Is a directory	126	
\$> \$HOMEdskjhfkdshfsd	ок ок	\$> \$>	0	
\$> "\$HOMEdskjhfkdshfsd"	ок ок	bash: : command not found \$>	127	
\$> '\$HOMEdskjhfkdshfsd'	ок ок	bash: \$HOMEsddfddfssfdfs: command not found \$>	127	
\$> \$DONTEXIST	ок ок	\$>	0	Cas d'un variable inexistant
\$> \$LESS\$VAR	ок ок	<pre>bash: -R: command not found \$&gt;</pre>	127	
\$> Ctlr-C	ок ок	\$> ^C \$>	130	
\$> holaCtlr-C	ок ок	<pre>\$&gt; hola^C \$&gt;</pre>	130	
<pre>\$&gt; cat (faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])</pre>			130	
	ок ок	^c \$>		
<pre>\$&gt; cat   rev(faire Ctlr-C apres avoir fait plusieurs fois [ENTREE])</pre>		<i>V</i>	0	
yestess ios (Emile-1)	ок ок	^c		
\$> ctlr-D	ок ок	^c \$> \$> exit	0	Sort du programme
() \$> hola Ctlr-D	ок <b>ок</b>	~/minishell \$> \$> hola	1	Ne fait rien
\$> Ctlr-\	ок ок	\$>	1	Ne fait rien
\$> hola Ctlr-\	ок ок	\$> hola	1	
<pre>\$&gt; cat (faire Ctlr-\ apres avoir fait plusieurs fois [ENTREE])</pre>	01/		131	
	OK OK	^\Quit (core dumped) \$>		
\$> Ctlr-Z	ок ок	\$>	1	Ne fait rien
<pre>\$&gt; sleep 3   sleep 3   sleep 3 (faire Ctlr-C une seconde apres)</pre>	ок ок	^c \$>	130	

<pre>\$&gt; sleep 3   sleep 3   sleep 3 (faire Ctlr-D une seconde apres)</pre>		ОК	ОК	\$>	0	Ne fait rien	
<pre>\$&gt; sleep 3   sleep 3   sleep 3 (faire Ctlr-\ une seconde apres)</pre>		ок	ок	^\Quit (core dumped) \$>	131		
\$> env		ок	ок	 PATH= S>	0		
\$> env hola	NON GERE	ок	ОК	env: 'hola': No such file or directory \$>	127		
\$> env hola que tal	NON GERE	ок	ОК	env: 'hola': No such file or directory \$>	127		
\$> env env	NON GERE	ОК	ок	 PATH= \$>	0	Affiche une seule fois env	
\$> env env env env	NON GERE	ок	ок	 PATH= \$>	0	Affiche une seule fois env	
\$> env ls	NON GERE	ок	ОК	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>	0		
<pre>\$&gt; env ./Makefile</pre>	NON GERE	ОК	ОК	env: './Makefile': Permission denied \$>	126		
<pre>\$&gt; export HOLA=bonjour \$&gt; env</pre>		ок	ок	 HOLA=bonjour \$>	0 0		
<pre>\$&gt; export HOLA=bonjour \$&gt; env</pre>		ок	ОК	 HOLA=bonjour \$>	0 0		
\$> export		ок	ОК	export PATH=""	0	Les variables sont triees par ordre alphabetique On n'oublie pas les doubles quotes qui designent les strings attribues a chaque variable	
\$> export Hola \$> export		ок	ОК	export HOME="/home/vietdu91" export Hola export PATH="" 5>	0 0	Hola n'existe pas dans env mais il existe dans export	
<pre>\$&gt; export Hola9hey \$&gt; export</pre>		ок	ОК	export HOME="/home/vietdu91" export Holadhey export PATH="" 5>	0 0		
\$> export \$DONTEXIST		ок	ок	export HOME="/home/vietdu91" export PATH="" S>	0	Le cas ou la variable n'existe pas et n'est pas importee	
\$> export   grep "HOME"		ок	ок	export HOME="/home/vietdu91" \$>	0		
\$> export ""		ОК	ок	<pre>bash: export: `': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export =</pre>		ок	ок	<pre>bash: export: '=': not a valid identifier \$&gt;</pre>	1		
\$> export %		ок	ок	bash: export: '%': not a valid identifier \$>	1		
<pre>\$&gt; export \$?</pre>		ок	ок	<pre>bash: export: `0': not a valid identifier \$&gt;</pre>	1		
<pre>\$&gt; export ?=2</pre>		ок	ок	<pre>bash: export: '?=2': not a valid identifier \$&gt;</pre>	1		
\$> export 9HOLA=		ок	ОК	bash: export: '9HOLA=': not a valid identifier \$>	1		
<pre>\$&gt; export HOLA9=bonjour \$&gt; env</pre>		ок	ок	 HOLA9=bonjour \$>	0		
<pre>\$&gt; export _HOLA=bonjour \$&gt; env</pre>		ок	ок	 _HOLA=bonjour \$>	0		
<pre>\$&gt; exportHOLA=bonjour \$&gt; env</pre>		ОК	ОК	HOLA=bonjour \$>	0		
<pre>\$&gt; export _H0_LA_=bonjour \$&gt; env</pre>		ОК	ок	 _HO_LA_=bonjour S>	0		
\$> export HOL@=bonjour		ок	ОК	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	
<pre>\$&gt; export HOL\~A=bonjour</pre>	NON GERE	ОК	ОК	<pre>s&gt; bash: export: `HOL~A=bonjour': not a valid identifier \$&gt;</pre>	1		

	\$> export -HOLA=bonjour			ОК	ОК	bash: export: -H: invalid option 2		
	\$> exportHOLA=bonjour				ок	bash: export:: invalid option 2		
	<pre>\$&gt; export HOLA-=bonjour</pre>				ок	bash: export: `HOLA-=bonjour': not a valid identifier 1		
	\$> export HO-LA=bonjour				ок	bash: export: `HO-LA=bonjour': not a valid identifier  \$>		
	\$> export HOL.A=bonjour				ок	bash: export: `HOL.A=bonjour': not a valid identifier \$>		
	<pre>\$&gt; export HOL\\\\$A=bonjour</pre>	NON GERE			ок	bash: export: `HOL\\$A=bonjour': not a valid identifier 1		
	<pre>\$&gt; export H0\\\\LA=bonjour</pre>	NON GERE			ок	bash: export: `HO\\LA=bonjour': not a valid identifier \$>		
	<pre>\$&gt; export HOL}A=bonjour</pre>				ок	bash: export: `HOL}A=bonjour': not a valid identifier 1		
	<pre>\$&gt; export HOL{A=bonjour</pre>				ок	bash: export: `HOL{A=bonjour': not a valid identifier \$>		
	\$> export HO*LA=bonjour				ОК	bash: export: `HO*LA=bonjour': not a valid identifier 1 \$>		
	\$> export HO#LA=bonjour				ок	bash: export: `HO#LA=bonjour': not a valid identifier 1 \$>		
	\$> export HO@LA=bonjour				ок	bash: export: `HO@LA=bonjour': not a valid identifier 1 \$>		
	<pre>\$&gt; export HO!LA=bonjour</pre>				ок	bash: !LA=bonjour: event not found 0		
	<pre>\$&gt; export HO\$?LA=bonjour \$&gt; env</pre>				ок	0 HODLA=bonjour \$>		
	<pre>\$&gt; export +HOLA=bonjour</pre>				ок	bash: export: `+HOLA=bonjour': not a valid identifier 1		
	\$> export HOL+A=bonjour				ок	bash: export: `HOL+A=bonjour': not a valid identifier 1		
	<pre>\$&gt; export HOLA+=bonjour \$&gt; env</pre>	NON GERE			ок	0 H01.A=bonjour \$>		
	\$> export HOLA=bonjour \$> export HOLA+=bonjour	NON GERE			ОК		"+=" s'ajoute au contenu de la variable	
	<pre>\$&gt; env  \$&gt; exportHOLA=bonjour \$&gt; env</pre>	NON GERE			ОК	<u>3&gt;</u>  PATH=		
	\$> export HOLA =bonjour				ОК	bash: export: '=bonjour': not a valid identifier 1		
	\$> export HOLA = bonjour			ОК	ОК	\$> bash: export: '=': not a valid identifier 1		
	\$> export HOLA=bon jour					\$> 0 HOLA=bon		
	\$> env				OK	\$>		
	<pre>\$&gt; export HOLA= bonjour \$&gt; env</pre>				ОК	0 HOLA= \$>		
	<pre>\$&gt; export HOLA=bonsoir \$&gt; export HOLA=bonretour \$&gt; export HOLA=bonjour \$&gt; env</pre>				ок	HOLA=bonjour 0 \$> 0		
	\$> export HOLA=\$HOME \$> env				ок	0 HOLA=/home/vietdu91 S > S		
	<pre>\$&gt; export HOLA=bonjour\$HOME \$&gt; env</pre>				ок	0 H01.A=bonjour/home/vietdu91 S>		
	\$> export HOLA=\$HOMEbonjour \$> env				ок	0 H0LA= 5>		
	<pre>\$&gt; export HOLA=bon\$jour \$&gt; env</pre>				ок	0 H01A=bon S>	Le cas où la variable "\$jour" n'existe pas	
Ь	<pre>\$&gt; export HOLA=bon\jour \$&gt; env</pre>	NON GERE			ок	0 H0LA=bonjour >>>		
UNSET	<pre>\$&gt; export HOLA=bon\\jour \$&gt; env</pre>	NON GERE			ок	0 HOLA=bon\jour 0 S>>		
ORT &	<pre>\$&gt; export HOLA=bon(jour</pre>	NON GERE		ОК	ок	bash: syntax error near unexpected token `('	Marche pour les caractères : ()	

\$> export HOLA="ls -l - a"		ОК	ОК	ls -l - a 0		
\$> echo \$HOLA="s -la" \$> export HOLA="s -la" \$> 15HOLA				\$> 0 total 16 0		
\$> L\$HOLA			ок	drwx 2 vietdu91 [] . drwxrwxewt 51 root []rwxr-xr-x 8 vietdu91 [] crashtest.c		
				[]		
<pre>\$&gt; export HOLA="s -la" \$&gt; l"\$HOLA"</pre>		ок	ок	bash: ls -la: command not found 127 \$>		
<pre>\$&gt; export HOLA="s -la" \$&gt; l'\$HOLA'</pre>		ок	ОК	bash: l\$HOLA: command not found 127		
\$> export HOLA="l" \$> \$HOLAs		ок	ОК	\$> 0		
<pre>\$&gt; export HOLA="l" \$&gt; "\$HOLA"s</pre>		ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c		
<pre>\$&gt; export HOL=A=bonjour \$&gt; env</pre>		ок	ОК	0 HOL=A=bonjour \$>		
<pre>\$&gt; export HOLA="l" \$&gt; '\$HOLA's</pre>		ок	ОК	bash: \$HOLAs: command not found \$>		
<pre>\$&gt; export HOL=A="" \$&gt; env</pre>		ок	ок	 H0L=A= \$>		
<pre>\$&gt; export TE+S=T \$&gt; env</pre>		ок	ОК	bash: export: `TE+S=T': not a valid identifier 1 \$>		
\$> export ""=""		ок	ОК	bash: export: `=': not a valid identifier 1 \$>		
\$> export ''=''		ок	ОК	bash: export: `=': not a valid identifier 1		
\$> export "="="="		ок	ОК	bash: export: `===': not a valid identifier 1 \$>		
\$> export '='='		ок	ОК	<pre>bash: export: `===': not a valid identifier \$&gt;</pre>		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; \$HOLA*BYE*d</pre>			ок	bash: pBYEd: command not found \$27		
\$> export HOLA=p \$> export SYE=w \$> "\$HOLA" '\$BYE'd		ок	ок	bash: p\$BYEd: command not found \$>		
\$> export HOLA=p \$> export SME=w \$> "\$HOLA""\$BVE'd		ок	ок	/home/vietdu91/42_works/minishell 0		
<pre>\$&gt; export HOLA=p \$&gt; export BYE=w \$&gt; \$"HOLA"\$"BYE"d</pre>		ок	ок	bash: HOLABYEd: command not found 127 \$>		
\$ export HDLA=p \$> export BYE=w \$> \$'HOLA'\$'BYE'd		ок	ОК	bash: HOLABYEd: command not found \$> 127		
\$> export HOLA=-n \$> "echo \$HOLA" hey		ОК	ОК	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"		ОК	ОК	1'2\$C"\$D5"\$F'7'8		
\$> export HOLA=bonjour \$> env \$> unset HOLA		ок	ОК	PATH= 0 Ps 0		
<pre>\$&gt; env \$&gt; export HOLA=bonjour</pre>				0		
\$> env \$> unset HOLA \$> unset HOLA			ок	PATH= \$>		
\$> env \$> unset PATH \$> echo \$PATH		ОК	OK	9		
\$> ecno \$PATH \$> unset PATH \$> 1s		ОК	OK	bash: ls: No such file or directory 127		
\$> Ls \$> unset ""		ОК	ОК	bash: unset: `': not a valid identifier 1		
\$> unset INEXISTANT		ОК	ОК	\$>	Cas où la variable n'existe pas	
\$> unset PWD \$> env   grep PWD \$> pwd		ок	ОК	\$> 0 \$> 0 /home/vietdu91/42_works/minishell 0		
				\$>		

\$> pwd			/home/vietdu91/42_works/minishell 0		
\$> unset PWD \$> env   grep PWD	Ok	ок	\$> 5>		
\$> cd \$PWD \$> pwd			\$> \\home/vietdu91 \qquad 0		
\$> unset OLDPWD \$> env   grep OLDPWD	Ok	ок	\$> \$> 0		
\$> unset 9H0LA	Ok	ОК	bash: unset: `9HOLA': not a valid identifier		
\$> unset HOLA9	Ok	ОК	\$> 1		
\$> unset HOL?A	Ok	ок	bash: unset: `HOL?A': not a valid identifier		
\$> unset HOLA HOL?A	OK	ОК	\$> 1 bash: unset: `HOL?A': not a valid identifier	Il unset bien HOLA	
\$> unset HOL?A HOLA			\$> 1 bash: unset: `HOL?A': not a valid identifier	Il unset bien HOLA	
	Ok	ок	\$> 1	It diset been nota	
\$> unset HOL?A HOL.A	Ok	ок	<pre>bash: unset: 'HOL?A': not a valid identifier bash: unset: 'HOL.A': not a valid identifier \$&gt; 1</pre>		
\$> unset HOLA=	ОК	ок	bash: unset: `HOLA=': not a valid identifier \$>		
\$> unset HOL\\\\A NON GERE	Ok	ОК	bash: unset: `HOL\\A': not a valid identifier		
\$> unset H0L;A NON GERE	Ok	ОК	bash: A: command not found		
\$> unset HOL.A	Ok	ОК	bash: unset: `HOL.A': not a valid identifier 1		
\$> unset HOL+A	Ok	ОК	bash: unset: `HOL+A': not a valid identifier		
\$> unset HOL=A	Ok	ОК	bash: unset: `HOL=A': not a valid identifier 1		
\$> unset HOL{A	OK	ок	bash: unset: `HOL{A': not a valid identifier		
\$> unset HOL}A	Ok	ОК	bash: unset: `HOL}A': not a valid identifier		
\$> unset HOL-A	Ok	ОК	bash: unset: `HOL-A': not a valid identifier		
\$> unset -HOLA	Ok		bash: unset: -H: invalid option 2		
\$> unset _HOLA	Ok	ОК	\$>		
\$> unset HOL_A	Ok	OK	\$>		
\$> unset HOLA_	Ok	OK	\$> 0		
\$> unset HOL*A	Ok	OK	bash: unset: `HOL*A': not a valid identifier		
\$> unset HOL#A	Ok	ОК	bash: unset: `HOL#A': not a valid identifier	73	
\$> unset \$HOLA	Ok	ок	\$>	Il va unset le contenu de \$HOLA, c'est-a-dire la variable bonjour	
\$> unset \$PWD	Ok	ок	<pre>bash: unset: `/home/vietdu91/42_works/minishell': not a valid identifier 1</pre>		
\$> unset HOL@	OK	ОК	bash: unset: `HOL@': not a valid identifier		
\$> unset HOL!A	Ok	ОК	bash: !A: event not found		
\$> unset HOL^A	Ok	ОК	bash: unset: `HOL^A': not a valid identifier 1		
\$> unset HOL\$?A	ОК	ок	\$>	Unset prend en compte le contenu de \$?, c'est-a- dire qu'il lit HOLOA ou HOL127A	
\$> unset HOL\~A	Ok	ОК	bash: unset: `HOL~': not a valid identifier 1		
\$> unset "" HOLA \$> env   grep HOLA	Ok	ок	bash: unset: `': not a valid identifier 1 \$> 0 \$>	HOLA sera quand meme unset !!	
\$> unset PATH \$> echo \$PATH	Ok	ок	\$>		
<pre>\$&gt; unset PATH \$&gt; cat Makefile</pre>	ОК	ок	bash: cat: No such file or directory 0 \$> 1		
\$> unset =	Ok	ОК	bash: unset: `=': not a valid identifier 0		
\$> unset =====	Ok		bash: unset: `======': not a valid identifier 0		
\$> unset +++++	Ok	ОК	bash: unset: `++++++': not a valid identifier 0		
\$> unset	Ok		\$> 0		
\$> unset export	Ok	ОК	\$> 0		
\$> unset echo	Ok		\$> 0		
\$> unset pwd	Ok	ОК	\$> 0		
\$> unset cd	Ok		\$>		
\$> unset unset	Ok	ОК	\$>		

	\$> unset sudo		ОК	ОК	\$>	0		
	\$> export hola   unset hola   echo \$?			ОК	0	0		
	\$> /bin/echo			ОК	*	0		
RS S	\$> /bin/echo Hola Que Tal		ОК	ОК	\$> Hola Que Tal	0		
불뿔	\$> /bin/env				\$>	0		
FIC				ок	 PATH= \$>			
_ 8	\$> /bin/cd Desktop			ок	bash: /bin/cd: No such file or directory	127		
HISTORIQUE	\$> history	NON GERE		ок	 558 echo \$? 559 clear 560 history \$>	Θ	Affiche les 500 dernières commandes	
HIS	\$> [touche du haut]			ок	\$> clear	1	Defile l'historique	
	\$> pwd			ок	/home/vietdu91/42_works/minishell \$>	0	Retourne \$PWD	
	\$> pwd hola			ок	/home/vietdu91/42_works/minishell \$>	0		
	\$> pwd ./hola			ок	/home/vietdu91/42_works/minishell \$>	0		
	\$> pwd hola que tal			ок	/home/vietdu91/42_works/minishell \$>	0		
	\$> pwd -p			ок	bash: pwd: -p: invalid option	2		
	\$> pwdp			ОК	bash: pwd:: invalid option	2		
	\$> pwdp		ОК	ОК	bash: pwd:: invalid option	2		
	\$> pwd p	NON GERE	ОК	ОК	/home/vietdu91/42_works/minishell	0		
	\$> pwd pwd pwd	NON OLKL	ОК	ОК	\$> /home/vietdu91/42_works/minishell	0	N'affiche qu'une fois	
	\$> pwd ls		ОК	ОК	\$> /home/vietdu91/42_works/minishell	0		
	\$> pwd ls env				\$> /home/vietdu91/42_works/minishell	О.		
	\$> cd			ОК	\$> e.	0	On doit se retrouver dans le dossier défini par	
	4- C0			ок			SHOME /home/vietdu91	
	\$> cd .			ок	\$>	0	L'user reste dans le même dossier /home/vietdu91/42_works/minishell	
	\$> cd ./			ок	\$>	0	/home/vietdu91/42_works/minishell	
	\$> cd ././.			ОК	\$>	0	/home/vietdu91/42_works/minishell	
	\$> cd ./././ \$> cd			ОК	\$>	0	/home/vietdu91/42_works/minishell	
				ок	\$>	U .	On se retrouve au dossier précédent /home/vietdu91/42_works	
	\$> cd/			ОК	\$>	0	/home/vietdu91/42_works	
	\$> cd/ \$> cd/.			OK	\$>	0	/home/vietdu91 /home/vietdu91/42_works	
	\$> cd/././.			ок ок	\$	0	/home/vietdu91/42_works	
	\$> cd srcs			ОК	\$>	0	/home/vietdu91/42_works/minishell/srcs	
	\$> cd srcs objs			ОК	bash: cd: too many arguments	1	/home/vietdu91/42_works/minishell	
	\$> cd 'srcs'			ОК	\$>	0	/home/vietdu91/42_works/minishell/srcs	
	\$> cd "srcs"			ок	\$>	0	/home/vietdu91/42_works/minishell/srcs	
	\$> cd '/etc'			ок	\$>	0	Marche aussi avec les double quotes	
	\$> cd /e'tc'			ок	\$>	0	/etc	
	\$> cd /e"tc"			ОК	\$>	0	/etc	
	\$> cd sr			ок	bash: cd: sr: No such file or directory \$>	1		
	\$> cd Makefile			ок	<pre>bash: cd: Makefile: Not a directory \$&gt;</pre>	1		
	\$> cd/minishell		ОК	ок	\$>	0	/home/vietdu91/42_works/minishell	

	\$> cd////		ОК	ОК	\$>	0	1	
	\$> cd/.///bin/ls			ОК	bash: cd:////bin/ls: No such file or directory S>	1	/home/vietdu91/42_works/minishell	
	\$> cd /			ОК	\$>	0	1	
	\$> cd '/'			ОК	\$>	0	Marche aussi avec les double quotes	
	\$> cd //	NON GERE	ОК	ОК	// <>	0	/	
	3> pwd	NON GERE		ок	<del>32</del> [/ 6-	0	Marche aussi avec les double quotes	
	\$> cd ///			ОК	1	0	1	
	\$> pwd \$> cd ///////		ОК	ок	/	0	/	
۵	\$> pwd \$> cd '//////' \$> pwd			ок	\$> /.	0	Marche aussi avec les double quotes	
P W	\$> pwd \$> cd /minishell			ок	bash: cd: /minishell: No such file or directory	1	I	
_	\$> cd /		ОК	ок	\$>	0	1	
8	\$> cd \$> cd _			ОК	bash: cd: _: No such file or directory	1		
	\$> cd -				<pre>\$&gt; /home/vietdu91/42_works/minishell</pre>	0	Retourne à \$OLD_PWD	
				ОК	\$>		\$OLD_PWD et \$PWD sont inversés /home/vietdu91/42_works/minishell	
	\$> cd	NON GERE		ок	\$>		Retourne à \$HOME \$OLD_PWD et \$PWD sont inversés /home/vietdu91	
	\$> cd			ОК	bash: cd:: invalid option \$>	2		
	\$> cd \$HOME			ОК	\$>	0	/home/vietdu91	
	\$> cd \$HOME \$HOME			ок	bash: cd: too many arguments \$>	1		
	\$> cd \$HOME/42_works			ок	\$>		/home/vietdu91/42_works	
	\$> cd "\$PWD/srcs"			ОК	\$>		/home/vietdu91/42_works/minishell/srcs	
	\$> cd '\$PWD/srcs'			ок	<pre>bash: cd: \$PWD/srcs: No such file or directory \$&gt;</pre>	1		
	\$> unset HOME \$> cd \$HOME			ок	bash: cd: HOME not set \$>	1		
	<pre>\$&gt; unset HOME \$&gt; export HOME= \$&gt; cd</pre>			ок	\$>	0	/home/vietdu91/42_works/minishell	
	\$> unset HOME \$> export HOME \$> cd			ОК	bash: cd: HOME not set \$>	1		
	\$> cd minishell Docs crashtest.c			ок	bash: cd: too many arguments S>	1		
	\$> cd /   echo \$? \$> pwd			ок	0 \$>	0	/home/vietdu91/42_works/minishell	
	\$> cd ~			ОК	\$>		/home/vietdu91	
	\$> cd ~/	NON GERE		ок	\$>	0	/home/vietdu91	
	\$> cd ~/   echo \$? \$> pwd	NON GERE		ок	0 \$>	0	/home/vietdu91/42_works/minishell	
	\$> cd *	NON GERE		ок	\$>		[BONUS] S'il n'y a qu'un seul <u>dossier</u> et rien d'autre	
		NON GERE		ок	<pre>bash: cd: minishell: No such file or directory \$&gt;</pre>	1	<b>[BONUS]</b> S'il n'y a qu'un seul <u>fichier</u> et rien d'autre	
	\$> cd *	NON GERE		ок	bash: cd: too many arguments \$>		[BONUS] S'il y au moins deux éléments	
	\$> mkdir a \$> mkdir a/b \$> cd a/b \$> rm -r//a \$> cd			ОК	chdir: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory \$>	1		
	<pre>S&gt; mkdir a S&gt; mkdir a/b S&gt; cd a/b S&gt; rm -r//a S&gt; pm</pre>		ок	ОК	pwd: error retrieving current directory: getcwd: cannot access parent directories: No such file or directory \$>	1		

	\$> mkdir a				/home/vietdu91/42_works/minishell/a/b	0		
	<pre>\$&gt; mkdir a/b \$&gt; cd a/b</pre>		ОК	ок	<pre>\$&gt; /home/vietdu91/42_works/minishell</pre>	0		
	\$> rm -r//a \$> echo \$PWD		OK	OK	\$>			
	\$> echo \$OLDPWD \$> mkdir a				/home	0		
	\$> mkdir a/b \$> cd a/b				<pre>\$&gt; /home/vietdu91/42_works/minishell/a/b</pre>	0 0		
	\$> rm -r//a \$> cd		ОК	ок	\$>			
	\$> echo \$PWD \$> echo \$OLDPWD							
	\$> mkdir a \$> cd a				//home/vietdu91/42_works/minishell/a	0		
	\$> rm -r/a \$> echo \$PWD		ОК	ок	//home/vietdu91/42_works/minishell	·		
	\$> echo \$OLDPWD				-	•		
	<pre>\$&gt; export CDPATH=/ \$&gt; cd \$HOME/</pre>	NON GERE	ОК	ок	\$>	0 0	A gerer ? /home	
	<pre>\$&gt; export CDPATH=/ \$&gt; cd home/vietdu91</pre>	NON GERE	ОК	ок	/home/vietdu91 \$>	0 0	A gerer ? /home/vietdu91	
	\$> export CDPATH=./ \$> cd .	NON GERE	ОК	ОК	\$>	0	A gerer ? /home/vietdu91/42_works/minishell	
	<pre>\$&gt; export CDPATH=./ \$&gt; cd</pre>	NON GERE	ОК	ок	\$>	0	A gerer ? /home/vietdu91/42_works	
	\$> chmod 000 minishell \$> ./minishell		ОК	ок	bash: ./minishell : Permission denied S>	126	,,	
	\$> ./minishell \$> ls hola		ОК	ок	ls: cannot access 'hola': No such file or directory	2		
	\$> ./Makefile		ОК	ок	<pre>\$&gt; bash: ./Makefile: Permission denied</pre>	126		
	\$> ./minishell				\$> \$>	0	Relance le Minishell	
			ОК	ОК			Il faut donc 2 fois exit pour exit	
	\$> env   grep SHLVL \$> ./minishell	NON CERE	ОК	014	\$>	0 0		
	\$> env   grep SHLVL \$> exit	NON GERE	OK	ОК		0		
	\$> env   grep SHLVL \$> touch hola \$> ./hola		ОК	ок	bash: ./hola: Permission denied	126		
	\$> ./hola \$> env "wc" -l		ОК	ОК	\$> 66	0		
SDS	\$> env "wc "-l				\$> bash: wc -l: command not found	127		
BÂTARDS	\$> expr 1 + 1		ОК	ОК	\$>	0		
m	· ·		ОК	ОК	2 \$>			
	\$> expr \$? + \$?		ОК	ОК	0 \$>	0	Fait la somme de la valeur de \$?	
	<pre>\$&gt; env -i ./minishell \$&gt; env</pre>		OK	ОК	PWD=/mnt/nfs/homes/emtran/Desktop/42_minishell SHLVL=1	0		
			OK	OK	_=/usr/bin/env \$>			
	<pre>\$&gt; env -i ./minishell \$&gt; export</pre>		OK-	OV	export OLDPWD export PWD="/mnt/nfs/homes/emtran/Desktop/42_minishell"	0		
			ОК	ОК	export PWD="/mnt/nfs/homes/emtran/Desktop/42_minishell" export SHLVL="1" \$>			
	\$> env -i ./minishell \$> cd		ок	ок	bash: cd: HOME not set \$>	1		
	\$> env -i ./minishell \$> cd ~	NON GERE	ОК	ОК	\$>	0	/home/vietdu91	
	\$> cd ~ \$> exit		ОК	ОК	exit	0		
	\$> exit exit				<pre>~/minishell \$&gt; exit</pre>	2		
			ОК	ОК	<pre>bash: exit: exit: numeric argument required ~/minishell \$&gt;</pre>			
	\$> exit hola		ОК	ок	exit	2		
	\$> exit hola que tal				bash: exit: hola: numeric argument required ~/minishell \$> exit	2		
			ОК	ок	ext: bash: exit: hola: numeric argument required ~/minishell \$>			
	\$> exit 42		ОК	ОК	exit ~/minishell \$>	42		
	\$> exit 000042		ОК	ок	exit	42		
	\$> exit 666		ок		~/minishell \$> exit	154		
			OK	ОК	exit ~/minishell \$>			

	\$> exit 666 666	01/		exit	1		
		ОК	ок	bash: exit: too many arguments \$>			
	\$> exit -666 666	ок	ок	exit bash: exit: too many arguments \$>	1		
	\$> exit hola 666	ок	ок	exit bash: exit: hola: numeric argument required ~/minishell \$>	2		
	\$> exit 666 666 666 666	ок	ок	exit bash: exit: too many arguments \$>	1		
	\$> exit 666 hola 666	ок	ок	exit bash: exit: too many arguments	1		
	\$> exit hola 666 666	ок	ок	exit bash: exit: hola: numeric argument required ~/minishell \$>	2		
	\$> exit 259	ОК	ок	exit ~/minishell \$>	3		
	\$> exit -4	ОК	ок	exit ~/minishell \$>	252		
	\$> exit -42	ок	ок	exit ~/minishell \$>	214		
	\$> exit -0000042	ок	ок	exit ~/minishell \$>	214		
	\$> exit -259	ок	ок	exit ~/minishell \$>	253		
	\$> exit -666	ок	ок	exit ~/minishell \$>	102		
	\$> exit +666	ок	ок	exit ~/minishell \$>	154		
	\$> exit 0	ок	ок	exit ~/minishell \$>	0		
	\$> exit +0	ок	ок	exit ~/minishell \$>	0		
	\$> exit -0	ок	ок	exit ~/minishell \$>	0		
-	\$> exit +42	ок	ок	exit ~/minishell \$>	42		
EXI	\$> exit -69 -96	ок	ок	exit bash: exit: too many arguments \$>	1		
	\$> exit666	ок	ок	exit bash: exit:666: numeric argument required ~/minishell \$>	2		
	\$> exit ++++666	ок	ок	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2		
	\$> exit ++++++0	ок	ок	exit bash: exit: ++++++0: numeric argument required ~/minishell \$>	2		
	\$> exit0	ок	ок	exit bash: exit:0: numeric argument required ~/minishell \$>	2		
	\$> exit "666"	ок	ок	exit ~/minishell \$>	154		
	\$> exit '666'	ок	ок	exit ~/minishell \$>	154		
	\$> exit '-666'	ок	ок	exit ~/minishell \$>	102	Marche aussi pour les double quotes	
	\$> exit '+666'	ок	ок	exit ~/minishell \$>	154	Marche aussi pour les double quotes	
	\$> exit '666'	ок	ок	exit bash: exit:666: numeric argument required ~/minishell \$>	2	Marche aussi pour les double quotes	
	\$> exit '++++666'	ок	ок	exit bash: exit: ++++666: numeric argument required ~/minishell \$>	2	Marche aussi pour les double quotes	
	\$> exit '6'66	ок	ок	exit ~/minishell \$>	154	Marche aussi pour les double quotes	
	\$> exit '2'66'32'	ок	ок	exit ~/minishell \$>	8	Marche aussi pour les double quotes	
	\$> exit "'666'"	ок	ок	exit bash: exit: '666': numeric argument required ~/minishell \$>	2		

\$> exit '"666"'			exit 2	
	ОК	ОК	bash: exit: "666": numeric argument required ~/minishell \$>	
\$> exit '666'"666"666	ОК	ок	exit/minishell \$>	
\$> exit +'666'"666"666	ок	ок	exit //minishell \$>	
\$> exit -'666'"666"666	ОК	ок	exit 86 ~/minishell \$>	
\$> exit 9223372036854775807	ОК	ок	exit 255 ~/minishell \$>	
\$> exit 9223372036854775808	ОК	ОК	exit bash: exit: 9223372036854775808: numeric argument required //minishell \$>	
\$> exit -9223372036854775808	ОК	ОК	exit 0 ~/minishell \$>	
\$> exit -9223372036854775809	ок	ОК	protective 2  exit bash: exit: -9223372036854775809: numeric argument required //minishell \$>	
\$> cat   cat   ls	ОК	ОК	Anticiant Open Dockers Drawings Makefile minishell 0 philosophers srcs test.c	Apparaît la ligne ls Et 3 retours à la ligne avec la touche Entrée
\$> ls   exit	ОК	ОК	\$>	Ne fait rien Chaque commande d'un pipeline est exécutée dans son propre sous-shell, qui est un processus
\$> ls   exit 42	ОК	ОК	\$> 42	distinct Ne fait rien
\$> exit   ls	ок	ОК	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c	
<pre>\$&gt; echo hola &gt; bonjour \$&gt; exit   cat -e bonjour</pre>	ОК	ОК	hola\$ 0	
\$> echo hola > bonjour \$> cat -e bonjour   exit	ОК	ок	\$> 0	
\$> echo   echo	ОК	ОК	0 S>	
\$> echo hola   echo que tal	ОК	ок	que tal 0	
\$> pwd   echo hola	ОК	ОК	hola 0 S>	
\$> env   echo hola	ОК	ОК	hola 0 S>	
\$> echo oui   cat -e	ОК	ОК	oui\$ 0	
\$> echo oui   echo non   echo hola   grep oui	ОК	ОК	\$>	
\$> echo oui   echo non   echo hola   grep non	ОК	ОК	\$> 0	
\$> echo oui   echo non   echo hola   grep hola	ок	ок	hola \$>	
\$> echo hola   cat -e   cat -e   cat -e	ОК	ок	hola\$\$\$ \$>	
\$> cd   echo "hola"	ОК	ок	hola 9	/home/vietdu91/42_works/minishell
\$> cd /   echo "hola"	ок	ок	hola 9	/home/vietdu91/42_works/minishell
\$> cd   pwd	ок	ок	/home/vietdu91/42_works/minishell 0 \$>	/home/vietdu91/42_works/minishell
\$> ifconfig   grep ":"	ок	ок	docker0: flags=4099 <up,broadcast,multicast> mtu 1500 0 ether 02:42:27:cf:95:49 txqueuelen 0 (Ethernet) [] \$\text{S}\$</up,broadcast,multicast>	
\$> ifconfig   grep hola	ОК	ОК	\$> 1	
\$> whoami   grep \$USER	ОК	ок	vietdu91	
<pre>\$&gt; whoami   grep \$USER &gt; /tmp/bonjour \$&gt; cat /tmp/bonjour</pre>	ок	ок	vietdu91 0 \$>	
<pre>\$&gt; whoami   cat -e   cat -e &gt; /tmp/bonjour \$&gt; cat /tmp/bonjour</pre>	ок	ок	vietdu91\$\$ \$>	
<pre>\$&gt; whereis is   cat -e   cat -e &gt; /tmp/bonjour \$&gt; cat /tmp/bonjour</pre>	ок	ок	ls: /usr/bin/ls /usr/share/man/man1/ls.1posix.gz 0 /usr/share/man/man1/ls.1.gz\$\$	

\$> ls   hola	ок ок	bash: hola: command not found 127
\$> ls   ls hola	ок ок	ls: cannot access 'hola': No such file or directory 2
\$> ls   ls   hola	ок ок	assis, help, compadent found
\$> ls   hola   ls	OK OK	\$> bash: hola: command not found 0
0. (0.1.)	ок ок	
\$> ls   ls   hola   rev	ок ок	bash: hola: command not found \$>
\$> ls   ls   echo hola   rev	ок ок	aloh \$>
\$> ls -la   grep "."	ок ок	total 16 drwx 2 vietdu91 [] . drwxrwxewt 51 root []rwxr-xr-x 8 vietdu91 [] crashtest.c [] S>
\$> ls -la   grep "'.'"	ок ок	\$> 0
<pre>\$&gt; echo test.c   cat -e  cat -e </pre>	ок ок	
\$> \s \	ок ок	crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$>
<pre>\$&gt; echo hola   cat   cat   cat   cat   grep hola</pre>	ок ок	hola \$>
\$> echo hola   cat	ок ок	hola 5>
\$> echo hola  cat	ок ок	hola 5>
\$> echo hola  cat	ок ок	
\$> echo hola cat	ок ок	hola 0
\$> echo hola    cat NON GERE	ок ок	S FRANCE
\$> echo hola     cat	OK OK	\$>
\$ celle liete	ок ок	
\$> ech o hola   cat	ок ок	bash: ech: command not found 127 bash: o: command not found 5>
\$> cat Makefile   cat -e   cat -e	ок ок	[] \$\$ PHONY: all clean fclean re\$\$
\$> cat Makefile   grep srcs   cat -e	21/	objs/%.o: srcs/%.c\$ 0 pb avec split_to_node lexer_expand.c 120 pb de lexer du cou
	ок ок	\$>
<pre>\$&gt; cat Makefile   grep srcs   grep srcs   cat - e</pre>	ок ок	\$>
<pre>\$&gt; cat Makefile   grep pr   head -n 5   cd file_not_exist</pre>	ок ок	\$>
\$> cat Makefile   grep pr   head -n 5   hello	ок ок	bash: hello: command not found 127 \$>
\$> export HOLA=bonjour   cat -e   cat -e	ок ок	5> 0
\$> unset HOLA   cat -e	ок ок	
<pre>\$&gt; export HOLA   echo hola \$&gt; env   grep PROUT</pre>	ок ок	\$>
\$> export   echo hola	ок ок	hola \$>
\$> sleep 3   sleep 3	ок ок	
\$> time sleep 3   sleep 3	ок ок	real0m3.002s user0m0.003s sys0m0.001s \$>
\$> sleep 3   exit	ок ок	\$> Le prompt apparaît au bout de 3 secondes
\$> exit   sleep 3	ок ок	\$> Le prompt apparaît au bout de 3 secondes

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

\$> ls && cat   echo				crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c	[BONUS]	
	NON GERE	ок	ок	philosophers srcs test.c		
				\$>		
\$> export "" && unset ""	NON GERE	ОК	ок	bash: export: `': not a valid identifier 1	[BONUS]	
	NON GERE	ок	ок		[BONUS]	
\$> (ls)	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell O philosophers srcs test.c S>	[BONUS]	
\$> ( ( ls ) )	NON GERE	ОК	ок	crashtest.c Docs Dockers Drawings Makefile minishell ophilosophers srcs test.c	[BONUS]	
\$> ( ( ) ls )	NON GERE	ОК	ОК	bash: syntax error near unexpected token `)'	[BONUS]	
\$> ls && (ls)				crashtest.c Docs Dockers Drawings Makefile minishell 0	[BONUS]	
	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c crashtest.c Docs Dockers Drawings Makefile minishell philosophers srcs test.c \$2 \$2 \$3 \$3 \$4 \$5 \$5 \$5 \$5 \$5 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6 \$6		
\$> (ls && pwd)	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c /home/vietdu91/42_works/minishell \$>	[BONUS]	
\$> ( ( ls&&pwd ) )	NON GERE	ОК	ок	crashtest.c Docs Dockers Drawings Makefile minishell O philosophers srcs test.c /home/vietdu91/42_works/minishell \$>	[BONUS]	
\$> ( ( ls ) &&pwd )	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c /home/vietdu91/42_works/minishell \$>	[BONUS]	
\$> (ls && ( ( pwd ) ) )	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell 0 philosophers srcs test.c /home/vietdu91/42_works/minishell \$>	[BONUS]	
\$> (Is && pwd) > hola \$> cat hola	NON GERE	ок	ок	crashtest.c Docs Dockers Dockers Drawings Makefile minishell philosophers srcs test.c /home/vietdu91/42_works/minishell \$5	[BONUS]	
\$> > hola ls && pwd	NON GERE	ок	ок	/home/vietdu91/42_works/minishell	[BONUS]	
\$> > hola (ls && pwd)	NON GERE	ОК	ОК	bash: syntax error near unexpected token `(' 2	[BONUS]	
\$> (> pwd) \$> ls	NON GERE	ок	ок	orashtest.c Docs Dockers Drawings Makefile minishell ophilosophers pwd srcs test.c	[BONUS]	
\$> (< pwd) \$> ls	NON GERE	ок	ок	\$> 0	[BONUS] Si pwd existe	
\$> (< pwd)	NON GERE	ОК	ОК	bash: pwd: No such file or directory	[BONUS] Si pwd n'existe pas	
\$> ( ( ( ( ( pwd) ) ) )				\$> /home/vietdu91/42_works/minishell 0	[BONUS]	
	NON GERE	ОК	ОК	\$>		
\$> () pwd	NON GERE	ОК	ОК	\$>	[BONUS]	
\$> > pwd (ls)	NON GERE	ОК	ок	bash: syntax error near unexpected token `(' 2 \$>	[BONUS]	
\$> (ls  pwd)&&(ls  pwd)	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c s-	[BONUS]	
\$> (lss  pwd)&&(lss  pwd)	NON GERE	ок	ок	bash: lss: command not found /home/vietdu91/42_works/minishell bash: lss: command not found /home/vietdu91/42_works/minishell \$>	[BONUS]	
\$> (lss&&pwd)&&(lss&&pwd)	NON GERE	ОК	ок	bash: lss: command not found \$27	[BONUS]	

	<pre>\$&gt; (ls &amp;&amp; pwd   wc) &gt; hola \$&gt; cat hola</pre>				crashtest.c Docs	0	[BONUS]	
	ar Cat Hota				Dockers Drawings Makefile	·		
		NON GERE	ок	ок	Makefile minishell philosophers			
					srcs test.c			
SES)	A (1- 00 and 1 an) a bala				1 1 34 \$>		Frontie 1	
(PARENTHESES)	\$> (ls && pwd   wc) > hola \$> (ls && pwd   wc) > hola \$> cat hola				crashtest.c Docs Dockers Drawings	0 0 0	[BONUS]	
AREN		NON GERE	ОК	ОК	Drawings Makefile minishell			
a d		NON GERE	OK	OK .	philosophers srcs			
					test.c 1 1 34 S>			
	\$> (ls && pwd   wc) >> hola \$> echo hey&&(ls && pwd   wc) > hola \$> cat hola				hey \$>	0	[BONUS]	
	<pre>\$&gt; cat hola</pre>				crashtest.c Docs []	Θ		
					srcs test.c			
		NON GERE	ок	ОК	1 1 34 crashtest.c Docs			
					[] srcs			
					test.c 1 1 34 \$>			
	\$> (pwd   wc) < hola	NON GERE	ок	ок	1 1 34 \$>	0	[BONUS]	
	\$> (ls && pwd   wc) < hola	NON GERE	ОК	ОК	crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c 1 1 34	0	[BONUS]	
	\$> (ls -z    pwd   wc) < hola				1 1 34 \$> ls: invalid option 'z'	0	[BONUS]	
	3> (ts -2    pwu   wc) < nota	NON GERE	ок	ок	1 1 34 \$>	·	[boungs]	
	<pre>\$&gt; echo hey &gt; hola \$&gt; (pwd   wc) &lt; hola</pre>	NON GERE	ок	ок	1 1 34 \$>	0	[BONUS]	
	<pre>\$&gt; echo hey &gt; hola \$&gt; (ls &amp;&amp; pwd   wc) &lt; hola</pre>	NON GERE	ок	ок	crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c 1 1 34	0	[BONUS]	
	\$> echo hey > hola				\$> ls: invalid option 'z'	0	[BONUS]	
	\$> (ls -z    pwd   wc) < hola	NON GERE	ОК	ок	1 1 34 \$>		[source]	
	\$> (ls -z    pwd && ls)	NON SERE	011	2"	ls: invalid option 'z' /home/vietdu91/42_works/minishell crashtest.c Docs Dockers Drawings Makefile minishell	0	[BONUS]	
		NON GERE	ОК	OK	crashtest.c Docs Dockers Drawlings Makerile minishell philosophers pwd srcs test.c \$>			
	\$> ls    (cat Makefile grep srcs) && (pwd wc)	NON GERE	ОК	ОК	crashtest.c Docs Dockers Drawings Makefile minishell philosophers pwd srcs test.c 1 1 34	0	[BONUS]	
		NON GERE	OK .	- OK	\$>			
	\$> ls -z && (ls) && (pwd)	NON GERE	ОК	ОК	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>				/home/vietdu91/42_works/minishell \$> crashtest.c	0 0 0	[BONUS]	
		NON GERE	ок	ок	Docs []			
					srcs test.c \$>			
	<pre>\$&gt; ls &gt; Docs/hey &amp;&amp; pwd &gt; hola \$&gt; cat hola \$&gt; cat Docs/hey</pre>				/home/vietdu91/42_works/minishell \$>	0 0 0	[BONUS]	
	\$> cat Docs/hey	NON GERE	ок	ОК	crashtest.c Docs []	0		
					ercs test.c \$			
					\$>			

\$> cd/ && pwd && pwd	SERE	ОК	ок	/home/vietdu91/ /home/vietdu91/ \$>	0	[BONUS]	
\$> (cd/ && pwd) && pwd	SERE		ок	/home/vietdu91/ /home/vietdu91/42_works/minishell \$>	0	[BONUS]	
\$> ls -z    cd//&&pwd \$> pwd NON C	SERE		ок	ls: invalid option 'z' /home/vietdu91 \$> /home/vietdu91 \$>	0 0	[BONUS]	
\$> ls -z    (cd//&āpwd) \$> pwd NON C	SERE		ок	ls: invalid option 'z' /home/vletdu91 \$> /home/vletdu91 \$> /home/vletdu91/42_works/minishell \$>	0 0	[BONUS]	
<pre>\$&gt; echo hola &gt; bonjour \$&gt; cat bonjour</pre>			ок	hola \$>	0 0		
<pre>\$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat bonjour</pre>			ок	hola que tal \$>	0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat &lt; bonjour</pre>			ок	\$> \$> hola que tal \$>	0 0 0		
<pre>\$&gt; echo hola &gt; bonjour \$&gt; rm bonjour \$&gt; echo que tal &gt;&gt; bonjour \$&gt; cat &lt; bonjour</pre>			ок	que tal \$>	0 0 0		
<pre>\$&gt; echo hola que tal &gt; bonjour \$&gt; cat bonjour</pre>			ок	hola que tal \$>	0		
<pre>\$&gt; echo hola que tal &gt; /tmp/bonjour \$&gt; cat -e /tmp/bonjour</pre>			ок	hola que tal\$ \$>	0 0		
<pre>\$&gt; export HOLA=hey \$&gt; echo bonjour &gt; \$HOLA \$&gt; echo \$HOLA</pre>			ок	\$> hola \$>	0 0 0	Ne fait rien	
<pre>\$&gt; whereis grep &gt; Docs/bonjour \$&gt; cat Docs/bonjour</pre>			ок	grep: /usr/bin/grep /usr/share/man/man1/grep.1.gz /usr/share/man/man1/grep.1posix.gz /usr/share/info/grep. info.gz \$>	0 0		
<pre>\$&gt; ls -la &gt; Docs/bonjour \$&gt; cat Docs/bonjour</pre>			ок	total 16 drwx 2 vietdu91 [] . druxrwxewt 51 root []rwxr-xr-x 8 vietdu91 [] crashtest.c []	0 0		
<pre>\$&gt; pwd&gt;bonjour \$&gt; cat bonjour</pre>			ок	/home/vietdu91/42_works/minishell \$>	0		
\$> pwd > bonjour \$> cat bonjour			ок	/home/vietdu91/42_works/minishell \$>	0		
\$> echo hola > > bonjour			ОК	bash: syntax error near unexpected token `>' \$>	2		
\$> echo hola < < bonjour			ОК	bash: syntax error near unexpected token `<' \$>	2		
\$> echo hola >>> bonjour			ок	bash: syntax error near unexpected token `>' \$>	2		
\$> > bonjour echo hola \$> cat bonjour			ок	hola \$>	0		
<pre>\$&gt; &gt; bonjour   echo hola \$&gt; cat bonjour</pre>			ок	hola \$> \$>	0 0		
<pre>\$&gt; prout hola &gt; bonjour \$&gt; ls</pre>			ок	bash: prout: command not found \$> bonjour []	127 0	Erreur mais file "bonjour" cree tout de meme	
\$> echo hola > hello >> hello >> hello \$> ls \$> cat hello			ок	5-> hello 5- hola 5-	0 0 0		
<pre>\$&gt; echo hola &gt; hello &gt;&gt; hello \$&gt; echo hola &gt;&gt; hello \$&gt; cat &lt; hello</pre>			ок	hola hola \$>	0 0 0		
> echo hola > hello >> hello >> hello > echo hola >> hello > echo hola > hello > ccho hola > hello >> hello > cat < hello		ок	ок	hola \$>	0 0 0 0		

	ок	ок	hola hola 9 \$>	
	ок	ок	hola	
	ок	ок	hola hola hola \$> 0	
	ок	ок	\$> 0 0	
	ок	ок	\$>	
	ок	ок	hola \$> \$> 0 0 0	
	ок	ок	hola	
	ок	ок	hola \$> \$> 0 0 0	
	ок	ок	hola	
	ок	ок	hola \$> \$> 0 0 0	
	ок	ок	hola \$> \$> 0 0	
	ок	ок	hola \$> \$> 0 0 0	
	ок	ок	hola \$> \$> 0 0	
	ок	ок	hola \$>	
	ок	ок	hola hola 5> 0 hey 6 \$> 0 \$> 0 \$>	
	ОК	ок	bonjour hola S>	
	ок	ок	\$>	
		ОК           ОК	ОК         ОК           ОК         ОК	OK OK STATE  OK OK

<pre>\$&gt; echo bonjo \$&gt; echo hello</pre>	ur > hola1 > hola2				hola1 hola2 []	0 0		
\$> echo 2 >ho	la1 >> hola2		ок	ок	\$> hello	0		
\$> ls \$> cat hola1 \$> cat hola2					netto 2	0		
\$> cat hola2	ur > hola1				\$> hola1 hola2 []	0		
\$> echo hello	ur > hola1 > hola2 ola1 > hola2				\$>	0		
\$> ls	otal > notaz			ОК	bonjour \$>	0		
<pre>\$&gt; cat hola1 \$&gt; cat hola2</pre>					2 \$>	0		
\$> > pwd \$> ls			ок	ОК	pwd S>	0		
\$> ts \$> < pwd			ОК	ОК	bash: pwd: No such file or directory	1		
			UK	OK	\$>			
<pre>\$&gt; &lt; Makefile</pre>	NON G	ERE		ОК	<pre>.: not enough arguments \$&gt;</pre>	1	Voir pourquoi	
\$> cat <pwd< td=""><td></td><td></td><td></td><td>ок</td><td>bash: pwd: No such file or directory</td><td>1</td><td></td><td></td></pwd<>				ок	bash: pwd: No such file or directory	1		
\$> cat <srcs <="" td=""><td>pwd</td><td></td><td></td><td>ок</td><td>bash: srcs/pwd: No such file or directory</td><td>1</td><td></td><td></td></srcs>	pwd			ок	bash: srcs/pwd: No such file or directory	1		
\$> cat <td>4</td> <td></td> <td></td> <td></td> <td>\$&gt; bash:/pwd: No such file or directory</td> <td>1</td> <td></td> <td></td>	4				\$> bash:/pwd: No such file or directory	1		
				ОК	\$>			
\$> cat >>				ок	<pre>bash: syntax error near unexpected token `newline' \$&gt;</pre>		Marche aussi pour les autres operateurs de redirection	
\$> cat >>>				ок	bash: syntax error near unexpected token `>'	2		
\$> cat >> <<				ок	bash: syntax error near unexpected token `<<'	2		
Â				UK	\$>	2		
\$> cat >> > >	> << >>			ОК	<pre>bash: syntax error near unexpected token `&gt;' \$&gt;</pre>	2		
<pre>\$&gt; cat &lt; ls</pre>				ок	bash: ls: No such file or directory \$>	1		
\$> cat < ls >	ls			ОК	bash: ls: No such file or directory	1		
\$> cat > ls1 ·	< 152				\$> bash: ls2: No such file or directory	1		
\$> ls				ок	\$> ls1 []	0		
Â					\$>			
\$> >>hola \$> cat hola				ок	\$>	0		
\$> echo hola :	> bonjour		ок	ОК	hola	0		
\$> cat < bonje \$> echo hola :				ОК	\$> hola	0		
\$> cat <bonjo< td=""><td>ur</td><td></td><td></td><td>UK</td><td>\$&gt;</td><td>0</td><td></td><td></td></bonjo<>	ur			UK	\$>	0		
\$> echo hola> \$> cat <bonjou< td=""><td>ſ</td><td></td><td></td><td>ОК</td><td>hola \$&gt;</td><td>0</td><td></td><td></td></bonjou<>	ſ			ОК	hola \$>	0		
\$> echo hola> \$> cat< bonjor				ок	hola \$>	0		
\$> echo hola	>bonjour			ок	hola	0		
\$> cat< \$> echo hola	bonjour > bonjour				\$> hola	0		
\$> cat	< bonjour			ОК	\$>	0		
<pre>\$&gt; cat &lt; srcs</pre>				ок	hola \$>	0		
\$> echo hola : \$> cat <srcs <="" td=""><td>&gt;srcs/bonjour</td><td></td><td></td><td>ок</td><td>hola \$&gt;</td><td>0</td><td></td><td></td></srcs>	>srcs/bonjour			ок	hola \$>	0		
	> bonjour al >> bonjour				hola	0		
\$> echo que to \$> cat < bonjo	al >> bonjour our			ОК	que tal \$>	0		
\$> echo hola : \$> rm bonjour	> bonjour				que tal	0		
\$> echo que ta	al >> bonjour			ОК	32	0		
\$> cat < bonje \$> e'c'"h"o he	our ola > bonjour			OK	hola	0		
\$> cat 'bo'"n	"jour			ОК	\$>	0		
∨ \$> ls	> bonjour\ 1 NON G	FRE	ок	ок	bonjour 1' [] \$>_	0		
\$> cat bonjou	r\ 1			- OK	hola \$>	0		
\$> echo hola :	> bonjour hey				bonjour []	0		
\$> cat bonjou	r			ок	hola hey	0		
\$> cat hey					<pre>\$&gt; cat: hey: No such file or directory</pre>	1		
V					\$>			

<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;srcs/bonjour &gt;srcs/hello <pre>\$&gt; cat srcs/bonjour srcs/hello</pre></pre>	ок	ок	bash: prout: No such file or directory 0 \$> 0 \$> 1
\$> cat srcs/bonjour srcs/hello \$> echo hola > srcs/bonjour \$> echo hey > srcs/hello			bash: prout: No such file or directory 0
<pre>S rm srcs/bonjour srcs/hello S&gt; srcs/bonjour &gt;srcs/hello <prout s=""> ls srcs</prout></pre>	ок	ок	S
<pre>\$ S- cat srcs/bonjour srcs/hello \$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/bonjour \$&gt; srcs/bonjour <pre>\$ srcs/hello \$&gt; srcs/bonjour <pre>\$ srcs/hello \$&gt; srcs/bonjour</pre></pre></pre>	ОК	ОК	bash: prout: No such file or directory  \$> \$> \$> \$1 bash: prout: No such file or directory  \$> \$> \$> \$> \$> \$> \$> \$> \$> \$> \$> \$> \$>
\$> cat srcs/hello			hey \$> bash: prout: No such file or directory  0
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; rm srcs/bonjour srcs/hello \$&gt; srcs/bonjour srcs/hello \$&gt; srcs/bonjour <pre>\$&gt; srcs/bonjour <pre>\$&gt; srcs/bonjour <pre>\$&gt; srcs/bonjour <pre>\$&gt; cat srcs/bonjour</pre>\$&gt; cat srcs/bonjour</pre></pre></pre></pre>	ок	ок	\$>
<pre>\$&gt; echo hola &gt;/bonjour \$&gt; echo hey &gt;/hello \$&gt; &gt;./bonjour &gt;/hello <prout \$=""> cat/bonjour/hello</prout></pre>	ОК	ок	bash: prout: No such file or directory \$> \$> \$1 0
<pre>\$&gt; echo hola &gt;/bonjour \$&gt; echo hey &gt;/hello \$&gt; rm/bonjour/hello \$&gt; &gt;/bonjour &gt;/hello <prout \$=""> ls</prout></pre>	ок	ОК	Dash: prout: No such file or directory 0 \$ \$bonjour hello 0 \$ \$ 1 6 6 7 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
<pre>\$&gt; cat/bonjour/hello \$&gt; echo hola &gt;/bonjour \$&gt; echo hey &gt;/hello \$&gt; &gt;/bonjour <pre>qrout &gt;/hello \$&gt; cat/bonjour</pre></pre>	ок	ОК	bash: prout: No such file or directory 8 \$> 0 \$> 1 hey 8>
<pre>\$&gt; echo hola &gt;/bonjour \$&gt; echo hey &gt;/hello \$&gt; rm/bonjour/hello \$&gt; &gt;/bonjour <pre>\$&gt; prout &gt;/hello \$&gt; ls \$&gt; cat/bonjour</pre></pre>	ок	ОК	bash: prout: No such file or directory  \$> bonjour  \$> 1 \$> 0  \$> 0  \$> 0  \$> 0  \$> 0  \$> 0  \$>
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; ssrcs/bonjour &gt;&gt;srcs/hello <prout \$=""> cat srcs/bonjour \$&gt; cat srcs/bonjour</prout></pre>	ОК	ок	bash: prout: No such file or directory 0 0 5 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour &gt;srcs/hello <prout \$=""> cat srcs/bonjour \$&gt; cat srcs/bonjour</prout></pre>	ОК	ок	bash: prout: No such file or directory 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;srcs/bonjour &gt;&gt;srcs/hello <prout \$=""> cat srcs/bonjour \$&gt; cat srcs/hello</prout></pre>	ок	ОК	bash: prout: No such file or directory  S 0 bonjour  S> 0 hey  S>
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; ssrcs/bonjour <pre>&gt; rost srcs/benjour</pre> \$&gt; cat srcs/bonjour</pre>	ОК	ок	bash: prout: No such file or directory 0 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour <pre>\$&gt; cat srcs/bonjour \$&gt; cat srcs/bello \$&gt; cat srcs/hello</pre></pre>	ок	ОК	bash: prout: No such file or directory  \$> bonjour  \$> 0 bnjour  \$> 0 bny  \$
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; &gt;&gt;srcs/bonjour <prout>&gt;srcs/hello \$&gt; cat srcs/bonjour \$&gt; cat srcs/hello</prout></pre>	ок	ОК	bash: prout: No such file or directory 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
<pre>\$&gt; echo hola &gt; srcs/bonjour \$&gt; echo hey &gt; srcs/hello \$&gt; eprout &gt;&gt;srcs/hello \$&gt; cat srcs/bonjour &gt;&gt;srcs/hello \$&gt; cat srcs/bello</pre>	ок	ОК	bash: prout: No such file or directory 8 S> 0 bonjour 1 S> 0 hey 8

<pre>\$&gt; echo hola &gt; bonjour \$&gt; echo hey &gt; hello \$&gt; <bonjour>hello \$&gt; cat bonjour \$&gt; cat hello</bonjour></pre>			ОК	ок	hola		
\$> echo hola > bonjour \$> echo hey > hello \$> >bonjour >hello < prout \$> cat bonjour \$> cat hello			ок	ок	bash: prout: No such file or directory 0 \$> 0 \$> 1 \$> 0		
\$> echo hola > bonjour \$> echo hey > hello \$> rm bonjour hello \$> >bonjour >hello < prout \$> cat bonjour \$> cat bello			ок	ок	bash: prout: No such file or directory  \$> 0 \$> 1 0		
\$> echo hola > bonjour \$> echo hey > hello \$> >bonjour <prout hello<br="">\$&gt; cat bonjour \$&gt; cat hello</prout>			ОК	ок	bash: prout: No such file or directory 0 \$> 0 \$> 1 hey 0 \$>		
S- echo hola > bonjour S- echo hey > hello S- rm bonjour hello S- >bonjour - prout hello S- cat bonjour			ОК	ок	bash: prout: No such file or directory 0 \$> 0 5> 0 1	Le fichier hello n'existe pas	
<pre>\$&gt; echo hola &gt; bonjour \$&gt; <bonjour cat="" wc=""  =""> bonjour \$&gt; cat bonjour</bonjour></pre>			ок	ок	0 0 0 0 5> 0		
<pre>\$&gt; rm -f bonjour \$&gt; rm bonjour &gt; bonjour \$&gt; ls -l bonjour</pre>			ок	ок	ls: cannot access 'bonjour': No such file or directory 0 0 2		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA \$&gt; ls</pre>			ок	ОК	'bonjour hello' [] 8 9 9 9		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;"\$HOLA" \$&gt; ls</pre>			ОК	ок	'bonjour hello' []		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$"HOLA" \$&gt; ls</pre>			ок	ок	HOLA [] 0 \$> 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;\$HOLA&gt;hey \$&gt; ls</pre>			ОК	ок	'bonjour hello' hey [] 8 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;hey&gt;\$HOLA \$&gt; ls</pre>			ок	ок	'bonjour hello' hey [] 0 \$> 0		
<pre>\$&gt; export HOLA="bonjour hello" \$&gt; &gt;hey&gt;\$HOLA&gt;hey&gt;hey \$&gt; ls</pre>			ок	ок	'bonjour hello' hey [] 0 \$> 0		
<pre>\$&gt; export A=hey \$&gt; export A B=Hola D E C="Que Tal" \$&gt; echo \$PROUTSB\$C &gt; /tmp/a &gt; /tmp/b &gt; /tmp/c \$&gt; cat /tmp/a \$&gt; cat /tmp/b \$&gt; cat /tmp/b</pre>			ок	ок	\$>		
!!! !!! !!!	Contenu du fich !!! Contenu du fich Contenu du fich	ier a : "Amo ier b : "012 ier c : "Pro	ur Tu es H 3456789" ut"	orrible"			
\$> <a <b="" <c<="" cat="" td=""><td></td><td></td><td>ОК</td><td>ок</td><td>Prout 9 \$&gt;</td><td></td><td></td></a>			ОК	ок	Prout 9 \$>		
\$> <a <b="" <c<br="" cat="">\$&gt; cat a \$&gt; cat b \$&gt; cat c</a>			ОК	ОК	\$> 0 \$> \$> Amour Tu es Horrible 0 0123436789 Prout \$>		
\$> >a ls >b >>c >d \$> cat a \$> cat b \$> cat c \$> cat d			ОК	ОК	\$> \$> \$> \$> 0 \$> Prout 0 \$> crashtest.c 0 ccs [] test.c \$>		

	\$> >a ls >b >>c >d \$> cat a \$> cat b \$> cat c \$> cat d	ок	ок	\$> 0 \$> 0 \$> 0 Prout 0 S> 0 Crashtest.c Documents [] test.c \$>	
	\$> echo hola > a > b > c \$> cat a \$> cat b \$> cat c		ок	\$> \$> hola \$>	
	<pre>\$&gt; mkdir dir \$&gt; ls -la &gt; dir/bonjour \$&gt; cat dir/bonjour</pre>		ОК	total 16 drwx 2 vietdu91 [] . drwx-rwxt 51 root [] []	
	\$> <a \$&gt; cat a</a 		ОК	Amour Tu es Horrible \$>	
	\$> >d cat <a>&gt;e \$&gt; cat a \$&gt; cat d \$&gt; cat e</a>		ок	\$> Anour 8- Anour 9 Anour 9 5- Anour	
	<pre>\$&gt; &lt; a &gt; b cat &gt; hey &gt;&gt; d \$&gt; cat d \$&gt; ls</pre>		ОК	Anour Tu Es Horrible	
	\$> cat << hola		ОК	> \$HOME	
	\$> cat << 'hola'		ОК	> \$HOME	
	\$> cat << "hola"		ок	> \$HOME	
	\$> cat << ho"la"		ок	> \$HOME	
	\$> cat << \$HOME		ОК	> prout > /home/vietdu91 > \$40 > \$40NE \$>-	
	<pre>\$&gt; cat &lt;&lt; hola &gt; bonjour \$&gt; cat bonjour</pre>		ОК	> prout > croutte > antoine pue > hola \$> prout croutte antoine pue	
U	\$> cat << hola   rev		ОК	> prout  > croutte > antoine pue > hola tuorp ettuorc eup eniotna \$>	
HEREDOC	\$> << hola		ОК	> chola > holache > hola > "hola" > 'hola' > hola \$ hola	
	\$> < <hola< td=""><td></td><td>ОК</td><td>&gt; &gt; &gt; hola \$&gt;</td><td></td></hola<>		ОК	> > > hola \$>	
	\$> cat <<	ОК	ок	bash: syntax error near unexpected token `newline' 2 \$>	

<pre>\$&gt; cat &lt;&lt; prout &lt;&lt; lol &lt;&lt; koala</pre>		OK	ОК	> prout > lol > koala \$>	0		
\$> prout << lol << cat << koala			ОК	> prout > lol > cat > koala bash: prout: command not found	127		
\$> << \$hola		ок	ок	> \$hola \$>	0		
\$> << \$"hola"\$"b"		ок	ок	> holab \$>	0		
\$> << \$"\$hola"\$\$"b"		ок	ок	> \$hola\$\$b \$>	0		
\$> << ho\$la\$"\$a"\$\$"b"		ок	ок	> hola\$a\$\$b > ho\$la\$a\$\$nb \$>	0		
\$> echo hola <<< bonjour			ок	hola \$>	0		
\$> echo hola <<<< bonjour		ок	ок	bash: syntax error near unexpected token `<' \$>	2		
\$> echo hola <<<< bonjour		ок	ок	bash: syntax error near unexpected token `<<' \$>	2		
\$> ls *	NON GERE	OK	ОК	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		ок	<pre>crashtest.c test.c \$&gt;</pre>	0	[BONUS]	
\$> ls *.hola	NON GERE	ок	ок	\$>	2	[BONUS]	
\$> cat M*le	NON GERE	ок	ок	[affiche le Makefile] \$>	0	[BONUS]	
\$> cat M*ee	NON GERE	ОК	O.K	cat: 'M*ee': No such file or directory	1	[BONUS]	
\$> cat Make*file	NON GERE		ок	[affiche le Makefile] \$>	0	[BONUS]	