COMMAND	BASH	GAUSHELL
BUILTINS SIMPLE		
echo	/n	OK
echo -n	return prompt	OK
echo -nnnnn	return prompt	OK
echo This is a test	This is a test\n	OK
echo -n This is a test	This is a testarepsa@	OK
echo -nnnn This is a test	This is a testarepsa@	OK
echo This is a test -nnnn	This is a test -nnnn\n	OK
echo \$?	Exit error	OK
env	prints environment variables	OK
env PATH	env: 'PATH': No such file or directory	ОК
ed	arepsa@c1r15s9:~\$	ОК
ed .	stay in the same directory	ОК
ed.	\nCommand 'cd.' not found, did you mean:	OK
od . Is	bash: cd: too many arguments	ОК
od//	arepsa@c1r15s9:/\$	OK
od /usr/bin	arepsa@c1r15s9:/usr/bin\$	OK
od nonexistent	minishell: cd: nonexistent: No such file or directory	OK
unset	return prompt	OK
unset nonexistentvariable	return prompt	OK
unset HOME	deletes HOME, returns prompt	ОК
cd (after unsetting HOME)	bash: cd: HOME not set	OK
unset USER	deletes USER from env	OK
export USER=bubu (do after unset)	adds to env new user bubu	OK
export =	bash: export: `=': not a valid identifier	ОК
export ABC=\$USER	changes ABC=username	ОК
export novalue	returns prompt, keeps "novalue" in export print, but do	esi OK
export=zzzzzz	returns promt, doesn't add any variables	ОК
export _	Export normally	ОК
fdgsdfhb=	returns prompt	ОК
unset blabla OLDPWD	deletes OLDPWD, returns prompt	ОК
unset ALL VARIABLES	unsets all, returns prompt	ОК
env (perform after unsetting all variables)	bash: env: No such file or directory	ОК
bwd	prints home, returns prompt	ОК
owd args	prints home, returns prompt	ОК
export	prints ordered env variables with "declare -x" before	OK
env	prints env variables	OK

avit	avita with avit and a 0	OK
exit	exits with exit code 0	
exit 300	exits with exit code 44	OK
exit -300	exits with exit code 212	OK
exit +300	exits with exit code 44	OK
exit 2 3	bash: exit: too many arguments (and does not exit)	OK
exit abc	bash: exit: abc: numeric argument required + exits progra	
exit 999999999999999999999999999999999999	bash: exit: 999999999999999999999999999999999999	
unset -n HOME	does nothing	OK .
QUOTES		
echo "Hello World"	Hello World	OK
echo "'Hello World'"	Hello World'	OK
"echo Hello World"	echo Hello World: command not found	OK
echo Hello World'	unclosed quotes	OK
echo "	unclosed quotes	OK
"/bin/ls"	execute Is	OK
./minishell	execute minishell	OK
"" s""	execute Is	OK
"""Is"""	ls': command not found	OK
""Is -I""	execute Is with details	OK
"e"'c"'h"o" Hello	Hello	OK
"pw"""d	prints home directory	OK
export USER=bubu LAST='quote'	updates user and creates LAST=quote	OK
echo '\$HOME e \$CASA e \$USER'	\$HOME e \$CASA e \$USER	OK
echo "\$HOME e \$CASA e \$USER"	(expande)HOME e e (expande)USER	OK
echo "\$HOME e '\$CASA' e \$USER"	(expande)HOME e \$CASA e (expande)USER	OK
echo "cama'	unclosed quotes	OK
echo "\$HOME echo '\$USER'	unclosed quotes	OK
"ls' 'wc"	ls' 'wc: command not found	OK
"Is" 'wc"	unclosed quotes	OK
ls grep "md wc -l	unclosed quotes	OK
ls grep ""md wc -l	0	OK
ls grep "md" wc -l	1	OK
echo "md" wc	113	OK
echo "\$HOME echo '\$USER'"	(expande)HOME echo '(expande)USER'	OK
echo ""carro' echo 'cama""	carro echo cama	OK
echo "fred' echo 'cama"	fred' echo 'cama	OK
echo "cat lol.c cat > lol.c"	cat lol.c cat > lol.c	OK
echo '\$USER'	prints \$USER	OK

export WWW=" -aaa "	saves variable WWW=" -aaa "	OK
export ZZ"Y=aaa"bb	exports variable ZZY="aaabb"	OK
cd ""	fica no mesmo diretorio	OK
unset MA"IL"	unsets MAIL	OK
echo 42' '42	42 42	OK
echo a'b'c'd'e'f'g'h'ï'j'k'l'm'n'o'p'q'r's't'	abcdefghijklmnopqrst	OK
echo \$USER'\$USER'	damachad\$USER	OK
export test="a b c"; echo \$test	a b c	OK
ls>"	No such file	OK
ls > \$nonexpansible	Ambiguous Redirect	OK
EXPANDER		
echo \$USER	prints username	OK
export ABC=\$USER	creates var ABC=username	OK
EXIT CODES		
cat nonexistant file or folder; echo \$?	exit code 1	OK
enter ./minishell; exit 300	exit code 44	OK
./not_executable_file	exit code 126	OK
< not_existing_file	exit code 1	OK
l echo 1	exit code 2	OK
Is not_existing_file	exit code 2	OK
PIPES		
"Is" 'wc'	execute Is and wc	OK
unset USER env grep USER	USER=arepsa (doesn't unset variable)	OK
unset OLDPWD Is	performs Is, doesn't unset variable	OK
echo USER export USER=bubu	echo doesn't perform, USER is not changed, returns p	ron OK
echo Hello World grep e	bash: syntax error near unexpected token ` '	OK
cat cat Is	performs Is and exits after 2 enters	OK
cat /dev/urandom head -c 100	returns 100 random characters	OK
REDIRECTS		
echo hello > outfile		OK
	ls: cannot access '>': No such file or directory	OK
ls ">" ola	ls: cannot access 'ola': No such file or directory	OK
< inacessible_file cat -e > output	creates empty output file	OK
<	bash: syntax error near unexpected token `newline'	OK
		I .

	bash: syntax error near unexpected token `newline'	OK
toot		OK OK
> test	creates empty output file	
< existingfile	Does nothing	OK
< testfile > testfile2	creates empty output file "testfile2"	OK
HEREDOCS		
<<	bash: syntax error near unexpected token `newline'	OK
>>	bash: syntax error near unexpected token `newline'	OK
cat << end wc -l	count lines of heredoc file	OK
cat << end grep e	returns heredoc lines that contain e	OK
<< end grep e	returns prompt	OK .
<< EOF cat	expands any variable inside it	OK
<< "EOF" cat	Does not expand	OK
<< 'EOF' cat	Does not expand	OK
<< "EOF' cat	unclosed quotes	OK
DENTRO de HEREDOCS		
\$USER e \$CASA e "\$ HOME" e '\$ HOME' "\$HOME"	(expande)USER e e "\$ HOME e '\$ HOME'	
'\$HOME' \$HOME	"(expande)HOME" '(expande)HOME' (expande)HOME	OK
\$CASA \$ CAMA	\$ CAMA	OK .
"\$CASA" "\$ CAMA" "\$USER" '\$HOME' \$USER \$ "\$ HOME" << HOME	"" "\$ CAMA" (espande)"USER" '(espande)HOME' (espande)USER \$ "\$ HOME" << HOME	0/4
HOWE << HOWE	(espande)05ER \$ \$ HOME << HOME	OK
00111111100		
COMMANDS		
ls \$jhrgsdhgbd	Executes Is	OK
unset PATH; Is	bash: Is: No such file or directory	OK
/dev/urandom	bash: /dev/urandom: Permission denied	OK
\$ <something does="" exist="" not="" that=""></something>	returns prompt	OK
TO SOLVE		
export test="ls -I"; \$test	runs Is -I	OK
11111	Command " not found, but can be installed with:	OK
<< eof; Ctrl + C	returns prompt	ОК
\$PWD	bash: <pwd>: Is a directory</pwd>	OK
cat /bin/ls		OK
./bin/lsg wc		OK
ping 8.8.8.8		OK
export test; export test=hello; echo \$test	hello	OK
пп	Command " not found	OK
"" Is	executes Is, and shows "command not found"	OK

./srcs (execute existing folder)	bash: ./srcs: Is a directory	OK	
a= ; echo \$?	exit code 0	OK	
export us\$er=blabla		OK	
unset weird*%chars		OK	
cd FILENAME WITH SPACE	bash: cd: too many arguments	OK	
echo hello > FILENAME WITH SPACE		OK	