```
#!/bin/bash

list_alldir(){
        echo directory" "$1   >>header.tmp
        ls -l $1 | tr -s ' '|grep "^d"|awk '{ print $8,$1,$5 }' >>header.tmp
        ls -l $1 | tr -s ' '|grep "^-"|awk '{ print $8 }' | while read LINE in
        do
            long=`expr $( cat $1/$LINE | wc -l )`
            debut=`expr $(cat body.tmp | wc -l ) + 1`
            cat $1/$LINE >>body.tmp
            ls -l $1/$LINE | tr -s ' '|awk '{ print nom,$1,$5,debut,long }' nom=$LINE debut=$debut
long=$long >>header.tmp
        done
        echo @ >>header.tmp
        for file in $1/*
        do
            if [ -d $file ]; then
            list_alldir $file
            fi
        done
}

init_del(){
    if [ -f path.tmp ]
    then
        rm path.tmp
    fi
    if [ -f header.tmp ]
    then
        rm header.tmp
    fi
    if [ -f archive ]
    then
        rm archive
    fi
    if [ -f body.tmp ]
    then
        rm body.tmp
    fi
    touch body.tmp
}

file_archive(){
    list_alldir "$1"
    first=`expr $( echo | wc -l header.tmp| cut -d " " -f1 ) + 3`
    echo "3:"${first} >> $2
    echo "" >> $2
    cat header.tmp >> $2
    cat body.tmp >> $2
    rm *.tmp
}

extrait(){
    IFS="\n"
    body_debut=`expr $( cat $1 | awk -F ":" ' NR==1 { print $2 }' ) - 1`
    cat $1 | awk 'NR==3,NR==line' line=$body_debut | while read LINE in
    do
        if [ $LINE != "@" ]
        then
            tmp1=$( echo $LINE | cut -d " " -f1 )
            tmp=$( echo $LINE | cut -d " " -f3 )
            if [ -z "$tmp" ]
            then
                if [ $tmp1 = "directory" ]
                then
                route=$( echo $LINE | cut -d " " -f2 )
                mkdir -p ./$route
                fi
            else
                dir_file_nom=$( echo $LINE | cut -d " " -f1 )
                droit=$( echo $LINE | cut -d " " -f2 )
                is_dir=$( echo ${droit:0:1} )
                u=$( echo ${droit:1:3} | sed -e 's/-//g' )
                g=$( echo ${droit:4:3} | sed -e 's/-//g' )
                o=$( echo ${droit:7:3} | sed -e 's/-//g' )
                if [ $is_dir = "d" ]
                then
                    mkdir -p ./$route/$dir_file_nom
                elif [ $is_dir = "-" ]
                then
                    touch ./$route/$dir_file_nom
                    tmp4=$( echo $LINE | cut -d " " -f5 )
                    if [ $tmp4 -gt 0 ]
                    then
                        tmp3=$( echo $LINE | cut -d " " -f4 )
                        line_debut=`expr $body_debut + $tmp3 `
                        line_fin=`expr $line_debut + $tmp4 - 1`
                        cat $1 | awk  'NR==debut,NR==fin' debut=$line_debut fin=$line_fin >> ./$route/
$dir_file_nom
                    fi
                fi
                chmod o=$o  ./$route/$dir_file_nom
                chmod u=$u  ./$route/$dir_file_nom
                chmod g=$g  ./$route/$dir_file_nom
            fi
        fi
    done
}
browse(){
IFS="\n"
body_debut=`expr $( cat $1 | awk -F ":" ' NR==1 { print $2 }' ) - 1`
path_racine=`expr $( cat $1 | awk 'NR==3' | awk ' { print $2 }' )`
cat $1 | awk 'NR==3' | awk ' { print $2 }' > path_now.tmp
echo -n "vsh:>"
while read LINE in
do
premier=$( echo $LINE | awk '{ print $1 }' )
deux=$( echo $LINE | awk '{ print $2 }' )
if [ "$deux" != "/" ]
then
    si_fin_s=$( echo $deux | grep "/$" )
    if [ -n $si_fin_s ]
    then
        deux=$( echo $deux | sed -e 's!/$!!' )
    fi
fi
case $premier in
pwd)
    path_now=`expr $( cat path_now.tmp )`
    if [ $path_racine = $path_now ]
    then
        echo "/"
    else
        path_dis=$( cat path_now.tmp | sed -e "s/^$path_racine//" )
        echo $path_dis
    fi
;;
exit)
    rm path_now.tmp
    break
;;
ls)
    line_now=3
    path_now=`expr $( cat path_now.tmp )`
    si_abs_path=$( echo ${deux:0:1} )
    if [ -z $deux ]
    then
        path_search=$path_now
    elif [ "$si_abs_path" = "/" ]
    then
        path_search=$path_racine$deux
    else
        path_search=$path_now/$deux
    fi
    cat $1 | awk 'NR==3,NR==line' line=$body_debut | cut -d " " -f2 | while read FILE in
    do
        line_now=`expr $line_now + 1`
        if [ $FILE = $path_search ]
        then
            echo "1" > dir_find.tmp
            cat $1 | awk 'NR==line,NR==line_fin' line=$line_now line_fin=$body_debut | while read DIR
in
            do
                if [ $DIR != "@" ]
                then
                    droit=$( echo $DIR | cut -d " " -f2 )
                    is_dir=$( echo ${droit:0:1} )
                    is_excu=$( echo $droit |  grep "x" )
                    if [ $is_dir = "d" ]
                    then
                        echo $DIR | awk ' { ORS=" ";print $1 "/" } '
                    else
                        if [ "$is_excu" = "$droit" ]
                        then
                            echo $DIR | awk ' { ORS=" ";print $1 "*" } '
                        else
                            echo $DIR | awk ' { ORS=" ";print $1 } '
                        fi
                    fi
                elif [ $DIR = "@" ]
                then
                    echo ""
                    break
                fi
            done
        else
            continue
        fi
    done
    if [ -f dir_find.tmp ]
        then
            rm dir_find.tmp
        else
            echo "le répertoire '$deux' n'existe pas"
        fi
;;
cd)
    path_now=`expr $( cat path_now.tmp )`
    if [ $deux = ".." ]
    then
        if [ $path_now = $path_racine ]
        then
            echo "vous etes dans le racine"
        else
            echo ${path_now%/*} > path_now.tmp
        fi
    elif [ $deux = "/" ]
    then
        echo $path_racine > path_now.tmp
    else
        si_abs_path=$( echo ${deux:0:1} )
        if [ $si_abs_path = "/" ]
        then
            path_entre=$path_racine$deux
        else
            path_entre=$path_now/$deux
        fi
        cat $1 | awk 'NR==3,NR==line' line=$body_debut | grep "^directory " |awk '{ print $2 }' |
while read dire in
        do
            if [ $dire = $path_entre ]
            then
                echo $dire > path_now.tmp
                echo "1" > file_find.tmp
                break
            fi
        done
        if [ -f file_find.tmp ]
        then
            rm file_find.tmp
        else
            echo "le répertoire $deux n'existe pas"
        fi
    fi
;;
cat | extract)
    line_now=3
    path_now=`expr $( cat path_now.tmp )`
    si_abs_path=$( echo ${deux:0:1} )
    si_rel_path=$( echo $deux | grep -n '/' )
    if [ -z $deux ]
    then
        echo "vous devez appliquer un nom de fichier"
    elif [ $si_abs_path = "/" ]
    then
        file_path=$( echo ${deux%/*} )
        path_search=$path_racine$file_path
        file_nom=$( echo ${deux##*/} )
    elif [ -z $si_rel_path ]
    then
        path_search=$path_now
        file_nom=$deux
    elif [ -n $si_rel_path ]
    then
        file_path=$( echo ${deux%/*} )
        path_search=$path_now/$file_path
        file_nom=$( echo ${deux##*/} )
    fi
    cat $1 | awk 'NR==3,NR==line' line=$body_debut | cut -d " " -f2 | while read FILE in
    do
        line_now=`expr $line_now + 1`
        if [ $FILE = $path_search ]
        then
            cat $1 | awk 'NR==line,NR==line_fin' line=$line_now line_fin=$body_debut | while read DIR
in
            do
                if [ $DIR != "@" ]
                then
                    droit=$( echo $DIR | cut -d " " -f2 )
                    is_dir=$( echo ${droit:0:1} )
                    if [ $is_dir = "-" ]
                    then
                        nom=$( echo $DIR | awk ' { print $1 } ' )
                        if [ $file_nom = $nom ]
                        then
                            debut=$( echo $DIR | cut -d " " -f4 )
                            debut_abs=`expr $debut + $body_debut `
                            long=$( echo $DIR | cut -d " " -f5 )
                            fin_abs=`expr $debut_abs + $long - 1 `
                            if [ $premier = "cat" ]
                            then
                                cat $1 | awk 'NR==de,NR==fi' de=$debut_abs fi=$fin_abs
                            else
                                cat $1 | awk 'NR==de,NR==fi' de=$debut_abs fi=$fin_abs > $file_nom
                            fi
                            break
                        fi
                    fi
                elif [ $DIR = "@" ]
                then
                    echo "le fichier $file_nom n'existe pas"
                    break
                fi
            done
        else
            continue
        fi
    done
```

```
;;
clear)
    clear
;;
help)
    echo "on propese les commandes suivant : pwd clear ls cd cat extract."
    echo "pour sortir le script, vous devez taper 'exit'"
;;
*)
    echo "parametre erreur, vous pouvez taper help"
;;
esac
echo -n "vsh:>"
done
}
#main
case $1 in
-archive)
    init_del
    if [ $# -ne 3 ]
    then
        echo "Parametre erreur"
    elif [ -f $2 ]
    then
        echo "Archive file existant, vous devez changer le nom ou supprimer le fichier"
    else
        file_archive "$3" "$2"
    fi
;;
-extract)
    if [ $# -ne 2 ]
    then
        echo "Parametre erreur"
    elif [ -f $2 ]
    then
        extrait "$2"
    else
        echo "Extract file n'existe pas"
    fi
;;
-browse)
    browse "$2"
;;
-help)
    echo "Pour archiver : -archive nom_archive nom_repertoire"
    echo "Pour extract  : -extract nom-archive"
;;
*)
    echo "Mauvais parametre, vous pouvez utliser parametre -help"
;;
esac
```