

```

#!/bin/bash
#=====
# SYNOPSIS
#   inverted_index - hashmap-like data structure
#
# AUTHOR
#   Mikaela Montaos
#
# DESCRIPTION
#   hashmap-like data structure
#
# EXAMPLE
#   See below
#
# RETURN VALUE
#   Return 0 if successful
#   Return 1 otherwise.
#
# FUTURE ENHANCEMENTS
#   None
#
# BUGS
#   None
#=====

#=====
#
# online help
#
#=====
help() {
    echo "
Description:
    service manipulation

Syntax:
    inverted_index -u <start_url> -l <number_of_levels>

Options:
    -u start_url
        recursively search the string from webpage and links mentioned in webpage
    -l number_of_levels
        how many levels to search into

Examples:
    $ inverted_index -u a.html -l 0
    student: {0}

```

```

$ inverted_index -u a.html -l 1
student: {0}
My: {1}
students: {1}
Your: {1}
teachers: {1}
"
    exit 0
}

#=====
#   main program
#=====
typeset start_url number_of_levels

if test $# -eq 0
then
    help
else
    start_url=
    number_of_levels=

    while [ "$#" -ne 0 ]
    do
        case "$1" in
            -u) shift; start_url="$1"; shift;;
            -l) shift; number_of_levels="$1"; shift;;

            *) echo "Error in $0: wrong option $1"; help; exit 1;;
        esac
    done
fi

#####
#
# Setup temporary file
#
#####
tmpname=${0##*/}
# if [ $(echo $tmpname | wc -c) -gt 8 ]
# then
#     typeset -L8 command=$tmpname
# else
#     command=$tmpname
# fi
# Keep the first 10 characters
command=`echo "$tmpname" | sed 's/^\(.....\).*/\1/'`

```

```

TEMPA=/tmp/${command}A$$
TEMPB=/tmp/${command}B$$
TEMPC=/tmp/${command}C$$
TEMPD=/tmp/${command}D$$
TEMPE=/tmp/${command}E$$
TEMPF=/tmp/${command}F$$
ALL_TEMP="$TEMPA $TEMPB $TEMPC $TEMPD $TEMPE $TEMPF"

```

```

trap "/bin/rm -f $ALL_TEMP; exit 1" \
    1 2 3 4 5 6 7 8 10 12 13 15 16 17 19 20 21
trap "/bin/rm -f $ALL_TEMP" 0

```

```

#####
#
# Processing
#
#####
# TEMPC stores all the URLs need to be processed at this level
echo "$start_url" > $TEMPC
while true
do
    if [ "$number_of_levels" = 0 ]
    then
        break
    fi

    # TEMPD is used to store the URLs to be processed at next level
    echo "" > $TEMPD

    # TEMPC stores all the URLs need to be processed at this level
    # Process one level
    cat $TEMPC | while read url
    do
        # echo current_level: $number_of_levels
        # echo $url

        lynx -dump "$url" > $TEMPA
        grep "$string" $TEMPA | while read line
        do
            echo "$url $line"
        done

        cat $TEMPA | sed -n '/^References/, $p' | sed '1,2d' | while read num
        next_url
        do
            echo $next_url
        done >> $TEMPD
    done
done

```

```
done

# Preparing for the next level
cp $TEMPD $TEMPC

(( number_of_levels = number_of_levels - 1 ))
# echo $number_of_levels
done
```

```
mva456@DESKTOP-OE7HLB3:~/cs522/inverted-index$ lynx "a.html"
```

a.html

```
student
a_1
a_2
```

a_1.html

```
<<<
My student
a_1_1
```

a_1_1.html

```
<<<
NPU student
NPU teacher
```

a_2.html

```
<<<=>=>
Your teacher
```