Module 5 assignment solution:


```
arjjun-g@arjjun-g-VMware-Virtual-Platform:~$ ls
assignment2   Desktop      Downloads   Music      snap
assignment3   dir1         file1       Pictures   Templates
assignment4   Documents    file2       Public     Videos
arjjun-g@arjjun-g-VMware-Virtual-Platform:~$ mkdir assignment5
arjjun-g@arjjun-g-VMware-Virtual-Platform:~$ ls
assignment2   assignment5   Documents   file2      Public      Videos
assignment3   Desktop       Downloads   Music      snap
assignment4   dir1          file1       Pictures   Templates
arjjun-g@arjjun-g-VMware-Virtual-Platform:~$ cd assignment5
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ nano file_analyzer.sh
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ chmod +x file_analyzer.
sh
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ ls -l
total 4
-rwxrwxr-x 1 arjjun-g arjjun-g 1837 Feb  4 19:41 file_analyzer.sh
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ mkdir testdir
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ nano testfile.txt
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ cat testfile.txt
Hello My name is Arjjun G
I'm Pursuing B.Tech Mechatronics Engineering at Puducherry Technological Univers
ity
I'm also focing on TinyML
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$
```

```
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ cd testdir
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5/testdir$ nano testfile2.t
xt
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5/testdir$ ls
testfile2.txt
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5/testdir$ cat testfile2.tx
t
The Future of TinyML is Tiny and Bright
The Tiny Part focuses on deploying ML Models on tiny systems called as embedded s
ystems
The Bright part focues on Low Cost, Low memory Computation, Small size, Low laten
cy and on board computing

arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5/testdir$
```

```
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5/testdir$ cd ..
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ ls -l
total 16
-rw-rw-r-- 1 arjjun-g arjjun-g   65 Feb  4 20:10 errors.log
-rwxrwxr-x 1 arjjun-g arjjun-g 1837 Feb  4 19:41 file_analyzer.sh
drwxrwxr-x 2 arjjun-g arjjun-g 4096 Feb  4 20:27 testdir
-rw-rw-r-- 1 arjjun-g arjjun-g  136 Feb  4 20:12 testfile.txt
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$
```

```
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ ./file_analyzer.sh -f te
stfile.txt -k Hello
Script name        : ./file_analyzer.sh
Total arguments    : 4
All arguments      : -f testfile.txt -k Hello
Last exit status   : 0
Hello My name is Arjjun G
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$
```

```
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$ ./file_analyzer.sh -d te
stdir -k ML
Script name        : ./file_analyzer.sh
Total arguments    : 4
All arguments      : -d testdir -k ML
Last exit status   : 0
Match found in: testdir/testfile2.txt
arjjun-g@arjjun-g-VMware-Virtual-Platform:~/assignment5$
```

file_analyzer.sh script:

#!/bin/bash

ERROR_LOG="errors.log"

log_error() {
   echo "[$(date)] ERROR: $1" | tee -a "$ERROR_LOG"
}

show_help() {
cat << EOF
Usage: $0 [OPTIONS]

Options:
 -d <directory>   Directory to search recursively
 -k <keyword>   Keyword to search
 -f <file>   Search keyword directly in a file
 --help   Display this help menu

Examples:
 $0 -d logs -k error
 $0 -f script.sh -k TODO
 $0 --help

```
EOF
}

recursive_search() {
    local dir="$1"
    local keyword="$2"

    for item in "$dir"/*; do
        if [ -f "$item" ]; then
            grep -q "$keyword" "$item" && echo "Match found in: $item"
        elif [ -d "$item" ]; then
            recursive_search "$item" "$keyword"
        fi
    done
}


DIR=""
KEYWORD=""
FILE=""


if [[ "$1" == "--help" ]]; then
    show_help
    exit 0
fi

while getopts ":d:k:f:" opt; do
    case "$opt" in
        d) DIR="$OPTARG" ;;
        k) KEYWORD="$OPTARG" ;;
        f) FILE="$OPTARG" ;;
        ?) log_error "Invalid option: -$OPTARG"; exit 1 ;;
        :) log_error "Option -$OPTARG requires an argument"; exit 1 ;;
    esac
done


echo "Script name     : $0"
echo "Total arguments   : $#"
echo "All arguments    : $@"
echo "Last exit status  : $?"


if [[ -z "$KEYWORD" ]]; then
    log_error "Keyword cannot be empty"
    exit 1
fi

if ! [[ "$KEYWORD" =~ ^[a-zA-Z0-9_]+$ ]]; then
    log_error "Keyword contains invalid characters"
```

```
    exit 1
fi

if [ -n "$FILE" ]; then
    if [ ! -f "$FILE" ]; then
        log_error "File does not exist: $FILE"
        exit 1
    fi

    grep "$KEYWORD" <<< "$(cat "$FILE")"
    exit 0
fi

if [ -n "$DIR" ]; then
    if [ ! -d "$DIR" ]; then
        log_error "Directory does not exist: $DIR"
        exit 1
    fi

    recursive_search "$DIR" "$KEYWORD"
    exit 0
fi

log_error "No valid options provided. Use --help"
exit 1
```

This Assignment is done by Arjjun G