1.

labinf18:~/ex1/test $ cp /export/home/stud/s234358/ex1/test/text01.txt /export/home/stud/s234358/ex1/test/text01

2.

labinf18:~/ex1/test $ ln text01 t1

3.

labinf18:~/ex1/bin $ ln /export/home/stud/s234358/ex1/test/t1 t2

4.

labinf18:~/ex1/bin $ cd ..

labinf18:~/ex1 $ ls -lR

.:

total 0

drwxr-xr-x 2 s234358 Domain Users 23 ott 5 11:12 bin/

drwxr-xr-x 4 s234358 Domain Users 42 ott 5 10:28 src/

drwxr-xr-x 4 s234358 Domain Users 95 ott 5 11:01 test/

./bin:

total 4

-rw-r--r-- 3 s234358 Domain Users 1868 ott 5 10:57 t2

./src:

total 0

drwxr-xr-x 2 s234358 Domain Users 6 ott 5 10:28 include/

drwxr-xr-x 2 s234358 Domain Users 6 ott 5 10:28 lib/

./src/include:

total 0

./src/lib:

total 0

./test:

total 12

drwxr-xr-x 2 s234358 Domain Users 6 ott 5 10:38 result/

drwxr-xr-x 2 s234358 Domain Users 6 ott 5 10:29 script/

-rw-r--r-- 3 s234358 Domain Users 1868 ott 5 10:57 t1

-rw-r--r-- 3 s234358 Domain Users 1868 ott 5 10:57 text01

-rw-r--r-- 1 s234358 Domain Users 1868 ott 3 13:53 text01.txt

./test/result:

total 0

./test/script:

total 0

5. the number of links associated to each file or directory represents the number of hard-links for that file or directory.

In the specific a hard-link is a pointer from that file or directory entry to the Inode. Moreover a leaf directory has

at least 2 hard links, the first one because it is pointed by itself, the second one because it is pointed by its parent

directory.

6.

labinf18:~/ex1 $ rm -r /export/home/stud/s234358/ex1/test/text01

rm: remove regular file '/export/home/stud/s234358/ex1/test/text01'? y

labinf18:~/ex1 $ cat /export/home/stud/s234358/ex1/bin/t2

```
################################################
#####                          #####
##          OPERATING SYSTEMS           ##
##                               ##
##             exercise 1             ##
##                               ##
##         student id: (xxxxxx)         ##
#####                          #####
################################################
```

This is line 12

This is line 13

This is line 14

This is line 15

This is line 16

This is line 17

This is line 18

This is line 19

This is line 20

This is line 21

This is line 22

This is line 23

This is line 24

This is line 25

This is line 26

This is line 27

This is line 28

This is line 29

This is line 30

This is line 31

This is line 32

This is line 33

This is line 34

This is line 35

This is line 36

This is line 37

This is line 38

This is line 39

This is line 40

This is line 41

This is line 42

This is line 43

This is line 44

This is line 45

This is line 46

This is line 47

This is line 48

This is line 49

This is line 50

This is line 51

This is line 52

This is line 53

This is line 54

This is line 55

This is line 56

This is line 57

This is line 58

This is line 59

This is line 60

This is line 61

This is line 62

This is line 63

This is line 64

This is line 65

This is line 66

This is line 67

This is line 68

This is line 69

This is line 70

This is line 71

This is line 72

This is line 73

This is line 74

This is line 75

This is line 76

This is line 77

This is line 78

This is line 79

This is line 80

This is line 81

This is line 82

This is line 83

This is line 84

This is line 85

This is line 86

This is line 87

This is line 88

This is line 89

This is line 90

This is line 91

This is line 92

This is line 93

This is line 94

This is line 95

This is line 96

This is line 97

This is line 98

This is line 99

This is line 100

It works because an hard-link is an association between an object name and its content, the directory-entry points to the

Inode, more directory-entries point to the same I-node and then the content of the Inode isn't lost until there is even

one remaining directory-entry pointing to it, thus the file is removed when the last of its hard links is removed.