Directory structure used by the Authentication Server (AS) operating at "tejo.tecnico.ulisboa.pt"

USERS/UID1 is the directory that contains authentication and control files for the user with UID1. 'UID1' directory name is exactly 5 digits long.

The directory named USERS belongs to the AS default directory.

USERS/UID1	-	directory for the user with UID1
/UID2	-	directory for the user with UID2
.	-	.
/UIDn	-	directory for the user with UIDn

Files that may be found in (under) each users directory at the AS (for instance USERS/UIDn):

A file named *UIDn_pass.txt* – This file contains the users password and is created when the user registers for the first time at the PD application. This file remains unchanged forever. It is deleted only if the user issues a 'remove' command at the *User* application. After the 'remove' command, the UIDn directory will be deleted as well.

A file named *UIDn_reg.txt* – This is a temporary file, which contains the PDIP and the PDPort. When the user unregisters, this file is deleted.

A file named *UIDn_login.txt* – This is a temporary file that exists when a given user is logged in through the *User* application. This file is deleted when the user logs out.

A file named *UIDn_tid.txt* – This is a temporary file that exists after the user validates a given operation. This file contains the TID, the validated operation and the name of the file to which the operation applies, if it is the case.

Directory structure used by the File Server (FS) operating at "tejo.tecnico.ulisboa.pt"

USERS/UID1 is the directory that contains the files stored for the user with UID1. 'UID1' directory name is exactly 5 digits long.

The directory named USERS belongs to the FS default directory.

Each users directory may contain at most 15 files.

File names may contain any alphanumeric character plus the symbols '_'; '-' and '.'. The maximum total length of any file name is 24 characters.

The users directory USERS/UIDn is created when the user uploads its first file. It is deleted if the user issues a successful 'remove' command at the *User* application.

To work with directory entries, students may follow and generalize the following example:

```
#include <dirent.h>   // Include the dirent.h header file
int ListDir(char *dirname) // A typical directory listing function.
{
    DIR *d;
    struct dirent *dir;
    d=opendir(dirname);
    if(d)
    {
        while((dir=readdir(d)) !=NULL)
        {
            printf("FILE: %s\n", dir->d_name);
        }
        closedir(d);
        return(1);
    }
    else
        return(-1);
}
```