## F28HS2 Hardware Software Interface Programming Laboratory 3

This laboratory involves building a command line navigation system for traversing rooms via linking doors.

A room has a string name, an array of up to 4 doors and a count of the number of doors:

struct room {char \* name; struct door \* doors[4]; int dp;};

A door has a string name for the room it connects to and a pointer to that room:

```
struct door {char * name; struct room * room;};
```

Room descriptions are held in a text file with the format:

```
Room1 name
Room1 Door1 name
....
*
...
RoomN name
RoomN Door 1 name
....
*
```

For example:

Hall Study Cellar Kitchen Study Hall Garden \* Cellar Hall \* Kitchen Hall Garden \* Garden Study Kitchen \*

Rooms are held in a stack of pointers to rooms:

```
#define MAX 10
struct room * rooms[MAX];
int rp;
```

with room count rp, initially 0.

There is a current room:

struct room \* r;

## Write functions:

A) i) struct door \* newDoor(char \* name) to create a new door called name with a NULL room pointer;

ii) struct room \* newRoom(char \* name) to create a new room called name with no doors;

iii) showRoom (struct room \* r) to display for room r the room name, and the names of the rooms the doors connect to on numbered lines;

- B) char \* readLine(FILE \* fin) to read a sequence of characters ending with a new line from a file pointer and return the corresponding string, or NULL at EOF;
- C) i) readRooms (FILE \* fin) to read room descriptions from a file pointer into the stack of pointers to rooms;
  - ii) showRooms () to show all the rooms in the stack of rooms;
- D) connect() to set the pointer for each door for each room in the stack of room by searching the stack for the room with the name associated with the door.

Write a program which reads and connects the rooms from a command line file, sets the current room to the first room in the stack and then repeatedly:

- i) displays the current room
- ii) prompts for and inputs a door number
- iii) sets the current room to the room associated with that door

## For example:

```
$ explore rooms.txt
Hall
1. Study
2. Cellar
3. Kitchen
enter door number> 3
Kitchen
1. Hall
2. Garden
enter door number>
```

HINT: Before you do anything else, draw a picture of the empty stack of rooms, and then draw, step by step, what happens as (a) room descriptions are read from the file, and (b) the rooms are connected together...!