Department of Computer Science & Engineering

BBM104 Introduction to Programming Laboratory

Experiment 1

EMRE HANCI - 21604552

Subject : Dynamic Arrays, Structs, File I/O

Submission Date : 28.02.2018

Due Date : 14.03.2018

Advisors : Dr. Gönenç Ercan, Dr. Öner Barut, Dr. Cumhur Yiğit Özcan, Dr. Ali Seydi Keçeli, R. A. Özge Yalçınkaya

-What was the problem?

In this assignment our advisors expect that to teach us "How can I use dynamic arrays? ", "How can I use dynamic struct?" and "How can I use file input-output functions?" with a board game program. Program is about two sides Heroes and Monsters, they have a position on board. The board has a lenght and height which is given into program from second input file with the loadmap command. The first input file (chars_<input_number>.txt) only for explain the characters like "<type>,<name>,<hp>,<damage>", the second input file (commands_<number>.txt) for processing the program according the command types. There are five kind of comman;
- LOADMAP: This command can be given like "LOADMAP <integer> <integer>" and board of the game has rows and collunms according the given integers.
-PUT: This command define the position of characters. The cell which is given into program must be carry the first letter of character's name. If there is an empty cell, it must carry '.'.
-SHOW: The expactation of this command is show map status or heroes status or monsters status according the second argument which is given with show commands.
-ATTACK: When this command given into program, the side which is given with attack command is attack other side, but there are some rules;
*Same type characters can not attack each other.
*Characters should be attack in an order according the chars_<input_number>.txt order.
*A character attack only neighbor cells.
*An attacked characters HP must be decrease the amount of damage values of attacking character.
*None of characters healty point can be less than 0.When healty point reach 0 of a character, it must be removed from map.
*If a hero kill a monster its xp should be increase one.
-MOVE: This command can be given only for heroes, for this command program should move the character new position from old position and assign the old position '.'.

-My solution

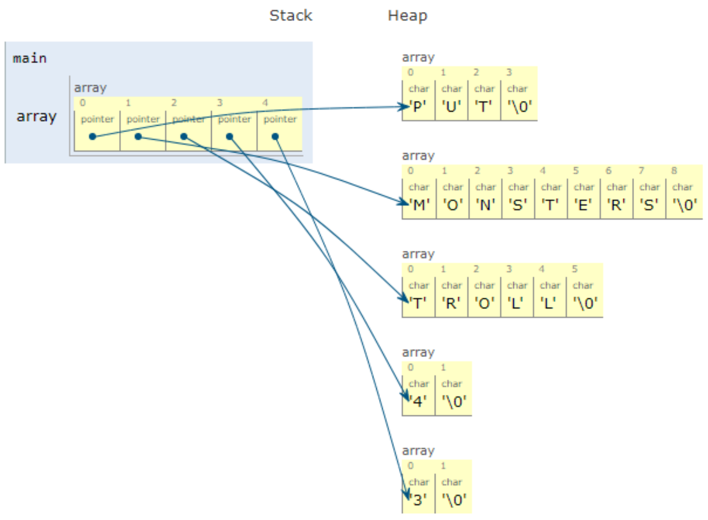
For the solve this problem I used six extra function. Firstly I open the chars file and the lines of file are counted by a for loop after get the number of characters, the file closed. Then I define a dynamic struct, I used one struct because the characters has same abilty the only diffrent is type because of that I add one attribute which is name type into my struct therefore I can access every characters in a struct.I open the chars file again for get the characters, after get the all characters, my work done on the chars file, then file closed. Then open the command file and get the text line by line. Every command has a differnt starting letter because of that I used switch-case and procces the line according the condition. After all commands executed the file closed, the area which is reserved from malloc is set free.

-The data structs, I used

I used one two dimension array which is created dynamicly and a struct which is also created dynamicly. The purpose of using two dimensinal array is about map. The map has row value and collumn value because of the map holds posititon of characters. The purpose of using struct is hold the data about characters. My struct has five variable (type,name,xp,hp,damage). The purpose of using the arrays and sturct dynamicly is access the data which is stored by arrays or struct.

-The explanation of my codes

As I mentioned in my solution part, I used six diffrent functiton now I will explain the purpose of those function and proccesing princeples.
Loadmap(char *line,int *row,int *col): Loadmap function for the define a dynamic two dimension array which is holds the status of map. I called function with the whole line which is begins with Loadmap command and send two integer variable which is holds the row and collumn values until the program terminated. Function split the line three after that define a dynamic two dimension array return it to main function.
Put (char **map,char *line): Loadmap functions purpose is put the characters into map I called functiton with the whole line which is begins with Put command and I send the map. In this command I can not known the amounts of characters therefore I set all the token of line into a array which is named "words". Now this arrays first two record holds "put" and "hero" or "monster" in here we do not need this two record therefore our map initilaze loop begin from two and increase three, beacuse after the first two records every three records has a relation for example:



Show (char **map,char *line,char *argv,struct characters character [ ], int x, int y, int Counter): The purpose of show function is show whole the map or show whole the monsters or heroes status. Line variable holds the whole line which is begin with Show command, map variable holds our map, if the command is show monster or show heroes, we need to know some information because of that I send struct to, the purpose of sending counter variable is about getting information from struct because counter holds the amount of struct size, and the purpose of x and y variable is about to stay in map because x holds row value y holds collumn value.
Move (char **map,char *line,char *argv,struct characters character [ ], int x, int y, int Counter): The purpose of move function is move heroes a place from another place. The variable same with show function variable they are holds same things. In this function we check three conditions, the conditions:
- If a place occupied character can not move there.
      For this condition we must check the new positon. For the check map we need x and y variables.
- If a place out of map character can not move there.
      For this condition we must check the new positon again. For the check map again we need x and y variables.
- Dead heroes can not move anywhere.
      For this condition we must check the struct because the moving character can be dead. For the check we need our main struct.

Attack (char **map,char *line,char *argv, int x, int y, int Counter, struct characters character[ ]): The purpose of this function is attack. It is most complicated part of program, the variables holds same things from the previous function. In this function firstly we check the types of characters, after that we need damage value for the attack other side therefore we gets the damage value from struct. Then we check the neighbor cells and if the neighbor cell is not empty we check the character is rival or friend. If it is rival, we decrease its healty point as much as the attacking characters damage. After that we check the attacked character dead or not, if it is dead we remove it from map. The last thing about attack is if a hero kill a monster its xp increase, therefore when heroes attack and a monster dead we increase the attacking characters xp.
Final (struct characters character [ ],int Counter,char *argv,char **map): The purpose of this function is check the all monsters or all heroes dead or not, if one of them true we should terminated the program, for the doing that we need the known all characters for that we send struct, the purpose of sending Counter variable is access the struct because its holds amount of records in struct. Then we check the struct if one of two sides dead, the program return 0 to main function. When its return 0 we set free the reserved places, close the files and terminate the program.

-The algorithm of my program

The algorithm of my codes;

- Output file opened for write purpose. (if it is not exist, created.)
- Chars file opened for read purpose, if it is not exist, program gives an warnings.
- The amount of characters taken from the file with a while loop.
- Chars file closed.
- Struct defines with dynamicly.
- Chars file opened for read purpose, if it is not exist, program gives an warnings.
- File taken line by line and split into variables.
- Variables get into struct.
- Counter increase.
- Chars file closed.
- Command file opened for write purpose, if it is not exist, program gives an warnings.
- Every command line taken into process line by line.
- If the start letter is ‘L’ program calls Loadmap() function.
    - Loadmap() creat a dynamic array and return it.
- If the start letter is ’P’ program calls Put() function.
    - Put() assing the cells accoring the command.
- If the start letter is ‘S’ program calls Show() function.
    - Show(), if show command given with map.

        Write map status to output file.
    - Show(), if show command given with monsters.

        Write monsters status to output file.
    - Show(), if show command given with heroes.

        Write heroes status to output file.
- If the start letter is ‘M’ program calls Move() function.
    - Move(), move heroes to new cell, if new cell empty otherwise it writes a message.
- If the start letter is ‘A’ program calls Attack() function.
    - Attack(), attack the other side of characters.
    - Final(), check all the charcters, if one of two sides dead, it return 0.
    - If Final() retrun 0, set free the memory, close the files and terminate the program.
    - Else does not things.
- If all the lines processed but neither monsters nor heroes dead, program set memory free close the files and terminate the program.