README

Project Description:

1. **mechanism of this game:**

```
while(console() != 'e')
{
        player_move (p3,p4,map,row,column);
}
```

   By the input is not e, player can move continue ,when player move a unit ,check encounter what, and check_move function is a bridge to other functions, then print the map , and thus game can be run.

Playing Method:

1. **How to play it:**
   Player can specify size of the map and the birth location ,then the game begin and the player use keyboard to play .
2. **Using what command to play:**
   Keyboard:

   ↑ ↓ ← →:move up , down , left , right
   A: (Attack)
   B: (Bag)
   L: (Level)
   U: (Level up)
   E: (Escape)
   1~9
3. **How does the game end?**
   When the player defeats all the monster , player wins and the game ends. The other method to end the game is press E .

Function Description:

intcreate_map (int*row , int*column);**

Make the player to specify the size of map , and this function can be use as a pointer point to the address of the map.

int check_boundary (int p_row , int p_column , int row , int column);

Check input row and column of birth location is in the map or not.

int check_availability (int p_row,int p_column,int row,int column,intmap);**

Check input row and column of birth location is in the empty space or not.

void monster_drop (int row , int column , intmap);**

Make the monsters be distributed in map randomly.

void boss_drop(int row , int column , intmap);**

Make the boss monster be distributed in map randomly.

void setup_player();

Let player specify his/her own birth location.

void setup_village (int row , int column , intmap);**

Make the village be distributed in map randomly.

void trap (int row , int column , intmap);**

Make the traps be distributed in map randomly.

void trap_decrease_blood ();

When the player touches the trap , blood of player will be deducted by a number.

void weapon_drop (int row,int column,intmap);**

The weapon will appear in the map with random number and the location.

void judge_win (intmap,int row,int column);**

By check whether there is any monster is survived.

int check_move (int p_row,int p_column,int row,int column,intmap);**

Check what player is encountered and the move of player is ruled or not.

void go_up (int*p_row,int*p_column,intmap,int row,int column);**

Make player move up.

void go_down (int*p_row,int*p_column,intmap,int row,int column);**

Make player move down.

void go_left (int*p_row,int*p_column,intmap,int row,int column);**

Make player move left.

void go_right (int*p_row,int*p_column,intmap,int row,int column);**

Make player move right.

void do_village ();

Let player decide what to do in the village

void choice_enter_village ();

Let player decide to enter the village or not.

void mall ();

Let player buy the weapon and the medicine.

void bag_judge ();

Show the things in the bag.

void gun ();

Make the player equip gun, and it will upgrade the attack.

void sword ();

Make the player equip sword, and it will upgrade the attack;

int*weapon_quip_index();

Save the index of the weapon which is equipped , when remove the weapon, make sure it will appear in the original location .

int*weapon_type_f();

Save the type of the weapon which is equipped, when remove the weapon, make sure it is the same weapon .

int monster_level_f();

Return monster level

int monster_blood_f(int monster_level);

Return monster blood

int monster_attack_f(int monster_level);

Return monster attack

int monster_defend_f(int monster_level);

Return monster defend

int monster_drop_experience (int monster_level);

Monster will drop experience , when the player is winning the monster.

int monster_drop_money (int monster_level);

Monster will drop money , when the player is winning the monster.

Because I do not use extern variable , I use function as a pointer to save the value of the player.

int*player_level_f();

Player level

int*player_strength();

Player attack

int*palyer_origin_attack();

Player attack without the weapon

int*player_defend_f();

Player defends

int*player_blood_f();

Player blood

int*player_max_blood_f();

Player blood ,because we have a location to heal ,I need a function save the blood without change.

int*money();

Player money , it can be used to buy things.

int*player_experience();

Player experience , it can be used to elevate level.

void look_level();

Press L , and player can look the value of attack , defend , blood , exp, level up need exp , money.

void level_up();

Press U, player can make the level up , and thus making attack , defend , blood be up.

int attack(int p_row,int p_column,intmap,int judge_boss);**

When the player encounters the monster or the boss monster ,player need to defeat them by making the monster blood be less than 0, and the monster will disappear. Then we can go to the space of monster.

int dodge ();

When player attack the monster or monster attack the player ,the dodge function will check the attack is successful or not. Return 1 : successful , 0 : fail.

int*bag_f ();

Bag has 25 spaces; we can put things in bag.

int*bag_index_f ();

When we want to add a new thing to bag , we should know what is the index of space to put.

int*player_weapon_equip_judge ();

Judge whether player has equipped one weapon or more.

void look_bag();

Press B , we can look what things are in the bag , and make a decision to use it or not.

void change_weapon();

When the game ask player what player want to do ,player can input 26 to remove the weapon which is equipped.

void add_warrior(int warrior);

Put things into the bag.

void Weapon ();

It will show what weapon the player can buy.

void Medicinal_liquid();

It will show what medicinal liquid(medicine) the player can buy.

void use_medicine();

Player use medicine and it will make the blood recover.

Variable Description

In the main function:

int row=0, column=0;

row,column save the sum row and sum column of the map.

int*p=&row ; int*p2=&column;

Pointer p,p2 can be sent to the functions to represent the map boundary.

int p_row=0 , p_column=0;

p_row , p_column save the row and column of player location which player specified.

int*p3=&p_row ; int*p4=&p_column;

pointer p3,p4 can be sent to function to represent the player location

In the other functions:

I will write beside the variable in the program ,because there is too many.

Version History

< 0 . 1 - Initial Release >