

Rapport du jeu sans bombe

Fonction main:

D'abord , je demande au joueur la taille de tableau et le but du jeu ,quand il répond, nous commençons le jeu .

Nous faisons l'initialisation du tableau . A condition que le tableau n'est pas full et il ne parvient pas le but ,nous faisons la suite :

Je demande au joueur la direction ,et tous les éléments se déplaceront et se confondront après la confirmation de direction . Et puis ,nous testons s'il est full et s'il parvient le but fixé avant ,s'il est full, il va afficher " Vous avez perdu la partie ! " . S'il parvient le but , il va afficher " Vous avez gagné la partie ! " .

Sinon, Nous continuons .

Quelques remarques:

Fonction maximum:

D'abord,on met `sup=tab[0][0]`(sup est le plus grand).Et puis ,on fait un parcours de tableau tab ,une fois on trouve une valeur supérieur de sup, on change la valeur de sup.

Fonction move_left:

D'abord ,tous les éléments dans le tableau se déplacent dans la direction gauche "largeur-1" fois , et puis ,s'il y a deux éléments de même valeur et ils sont adjacents dans la ligne d'horizon ,la nouvelle valeur de gauche est la somme des deux valeurs ,et la nouvelle valeur de droite est 0(0==NULL).

Et finalement ,encore une fois ,tous les éléments dans le tableau se déplacent dans la direction gauche "largeur-1" fois.

Fonction move_right:

D'abord ,tous les éléments dans le tableau se déplacent dans la direction droite "largeur-1" fois , et puis ,s'il y a deux éléments de même valeur et ils sont adjacents dans la ligne d'horizon ,la nouvelle valeur de droite est la somme des deux valeurs ,et la nouvelle valeur de gauche est 0(0==NULL).

Et finalement ,encore une fois ,tous les éléments dans le tableau se déplacent dans la direction droite "largeur-1" fois.

Fonction move_up:

D'abord ,tous les éléments dans le tableau se déplacent dans la direction haut "hauteur-1 " fois , et puis ,s'il y a deux éléments de même valeur et ils sont adjacents dans la ligne verticale ,la nouvelle valeur de haut est la somme des deux valeurs ,et la nouvelle valeur de bas est 0(0==NULL).

Et finalement ,encore une fois ,tous les éléments dans le tableau se déplacent dans la direction haut "hauteur-1" fois.

Fonction move_down:

D'abord ,tous les éléments dans le tableau se déplacent dans la direction bas "hauteur -1" fois , et puis ,s'il y a deux éléments de même valeur et ils sont adjacents dans la ligne verticale ,la nouvelle valeur de bas est la somme des deux valeurs ,et la nouvelle valeur de bas est 0(0==NULL).

Et finalement ,encore une fois ,tous les éléments dans le tableau se déplacent dans la direction bas "hauteur-1" fois.

Fonction ajouter:

D'abord ,on cherche une espace vide aléatoirement ,et puis on le met 1.

Fonction Isfull:

Tester si le tab est full et s'il devient impossible de faire evoluer la situation,si c'est vrai, retourne 1 et 0 sinon

Fonction parvient:

On fait un parcours dans le tableau et on cherche la valeur supérieure ou égale de "but" ,retourne 1 si on l'a trouvé et 0 sinon .

Instruction:

D'abord ,on vous demande la taille de tableau et votre "but ".

Et puis ,on vous demande la direction ,si vous entrez la lettre correctement ,le tableau changera après votre confirmation de direction ; sinon ,il y aura des instructions pour vous aider .

Jusqu'à un moment où vous avez eu une valeur supérieur ou égale au "but", bravo , vous gagnez la partie! Ou jusqu'à un moment où le tableau est full et il devient impossible de faire evoluer la situation , dommage , vous avez perdu la partie .