

## PG 110: PROJET BOMBEIRB

2016-2017 Selmi Marouen – Burguet Paul

### I- Gestion des déplacements

Pour gérer les déplacements, nous avons modifié la fonction `player_move` et `player_move_aux`, ajouté des conditions pour que le joueur ne dépasse pas la map.

### II-Gestion des mondes

Pour passer d'un monde à l'autre, nous avons créé une fonction `load_map` qui lit un fichier texte pour charger les maps. À chaque fois que le joueur passe par une porte ouverte (à condition qu'il a suffisamment de clés) nous incrémentons le niveau actuel du joueur et dans `game.c` nous récupérons ce niveau (dans `game_update`) pour charger le prochain monde. Dans le jeu, il y a 4 mondes dont un qui s'affiche si le joueur sauve la princesse.

Pour gérer le panneau d'information, nous décalons les numéros dans `game_banner_display`. Le problème qui se pose c'est que nous avons pas réussi à afficher le drapeau et le niveau actuel du joueur. Et il n'y a pas de portes qui permettent de revenir dans le monde précédent.

### III-Gestion des bombes

Pour gérer les bombes, nous avons utilisé une liste chaînée. Dedans, chaque bombe pointe vers la bombe suivante que le joueur a posée.

Quand le joueur appuie sur espace, le nombre de bombes du joueur se décrémente, nous ajoutons une bombe dans la liste des bombes initialement nulle, et enfin nous posons une bombe dans la map. Pour que la bombe explose, nous utilisons un compteur qui récupère le temps actuel du jeu et à chaque fois que la différence entre le temps actuel et 0 dépasse une seconde, on change l'état de la bombe à l'aide de la fonction `bomb_state`. Dès que la bombe est à l'état d'explosion, elle explose et le feu s'affiche en croix à l'aide de la fonction `bomb_display` et `bomb_list_display`.

Quand un monstre est touché, il disparaît. Quand le joueur est touché par une explosion, il perd une vie. Les caisses explosent, et à la place des caisses nous affichons les bonus ou malus. Le problème posé dans cette partie, c'est que quand le joueur ne bouge pas et la bombe explose il perd une vie, par contre, quand le joueur se dirige vers l'explosion, il perd pas de vie. L'affichage des bombes dépasse parfois les limites de la map. Quand le joueur n'a plus de vie, le jeu quitte automatiquement.

### IV-Gestion des monstres

Pour la gestion des monstres, c'est à peu près pareil que le joueur pour les mouvements, et comme les bombes pour les mouvements au hasard. Quand le monstre touche un joueur, ce dernier perd une vie et vice versa. Chaque map a ses propres monstres. Et à chaque fois que nous passons d'un monde à l'autre, la vitesse des monstres augmentent.. Normalement, pour les monstres, il n'y a aucun problème.

### V-Fin de partie

Si le joueur perd toutes ses vies, le jeu quitte automatiquement. Par contre, quand le joueur récupère la princesse, nous affichons une map avec YOU WIN dedans fait avec les éléments de décor de la map (arbres et pierres).

### VI- Pause

Quand le joueur, appuie sur [P], le jeu se met en pause grâce à une variable `pause` dans `game.c` initialisé à 0. Quand le joueur appuie sur [P], la variable devient 1 et nous mettons une condition

pour les mouvements des monstres, du joueur et de l'explosion de la bombe dans game\_update.

### VII-Sauvegarde

Nous n'avons pas réussi à faire cette partie.