

serveur

note: methode chargement de labyrinthe \Rightarrow n° fantômes

chargement du labyrinthe

- Labyrinthe crée à la main
- (A11-0) algorithme pour générer un labyrinthe aléatoire

prétraitement du partie

- placement des fantômes (aléatoire sur les cases libres)
- association d'une adresse IP et un port multi diffusion (communiquer le mouvement des fantômes etc)

annonceur du debut

- diffusion message [START***]
- diffusion message [WELCOMU h u l p u port***]
- m: numero de partie
- h: hauteur de partie
- w: largeur de partie
- P: n° fantômes
- 'P: adresse multi diffusion
- port: port multi diffusion
- envoie un message par chaque joueur
- [POSIT id u x y***]
- id: identifiant de joueur
- x: numero de ligne de debut, $0 \leq x \leq w-1$
- y: numero de colonne de debut $0 \leq y \leq h-1$

GLIS? ***

- renvoyer [GLIS!_s]
- s: nb de joueurs de la partie
- s fait:
- renvoyer [GLYR id u x y u p***]
- id: id du joueur
- x y: position de ce joueur
- p: nb des points

recevoir commande client

commandes possibles:

- Up mou u l***
- Down u l***
- Left u l***
- Right u l***
- Iquit ***

GLIS?***

MALL?_l mess

SEND?_l id u mess

id: identifiant de destinataire

Mess: message

IQUIT ***

- renvoyer [GOBYE ***]
- fermer connexion avec joueur

tout les joueurs quittent

termination de partie

déconnexion des joueurs

- reçoit n'importe quelle message
- renvoyer [GOBYE ***]
- fermer la connexion

MALL?_l mess

- mess serait diffuse sur le port et l'adresse IP
- renvoyer [MALL! ***]

SEND?_l id u mess

verifier id

id valide pour la partie

message autorisé

- envoie du mess au joueur id
- renvoyer [SEND! ***]

message refusé

- renvoyer [NSEND ***]

mouvement

- déplacer dans direction en tant que d=p ou on voit un mur
- P++ chaque iteration
- R: cas de plus

mur

fantôme?

renvoie [MOVE! x y u p***]

renvoie [MOVE! x y u p***]

renvoie [MOVE! x y u p***]

renvoie [MOVE! x y u p***]