- 1 labyrinthe aléatoire par niveau (6 niveaux ou plus si le temps nous le permet)
- « Pour une meilleure expérience de jeu, mettez votre terminal en plein écran »

## Labyrinthe:

Fonctions de vérification de la map : Couloirs donnent sur des pièces Couloirs ne se croisent pas Toutes les pièces communiquent Taille des pièces correctes

Intégrées dans les fonctions de création

Création de pièces (nombre augmente avec le niveau)
Création des couloirs

Dans une matrice qui représentera la map entière délimitée par des caractères

Placement des trésors

Entrée et Sortie fixe (facilite la création des salles et ds couloirs)

Contour de la map

# Gestion déplacements :

Déplacement du perso dans l'environnement

Couloirs non pris en compte (on passe d'une porte à la porte à l'autre bout du couloir) :

Quand on arrive sur un « p », si au prochain déplacement nous avons un « # » qui correspond à un couloir, on recherche dans les alentours de ce « # » s'il y a un « # » cela jusqu'au prochain « p » sur lequel nous placerons le joueur.

Refresh (a chaque mouvement : moins fluide mais meilleure complexité) Hitbox (possibilité de déplacement ou pas notamment dans les murs)

#### Décompte de temps :

Utilisation des fonctions éventuelles (time.h)

## Légende sur le jeu :

Légende claire (évolutive si monstres ou pièges)

# Compteur de trésors :

Simple fonction de comptage (avec incrémentation d'un compteur)

Romane Lucien <mark>Yann</mark> A déterminer