

Sujet *Le jeu de quête des années 80*

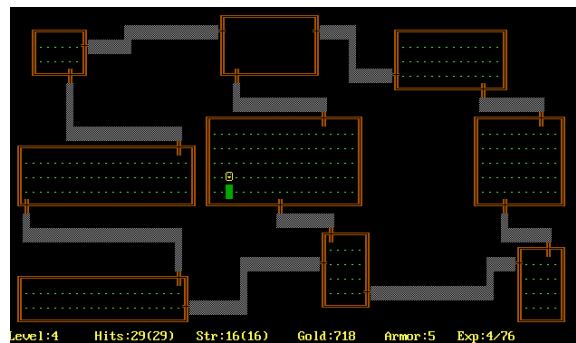
UE12 informatique - Apprentissage de la Programmation - année 2025-2026
Hackathon du 30 janvier 2026

Inspirez-vous librement du très célèbre jeu de parcours de donjons **Rogue** pour proposer votre propre version du jeu !

Vous ne connaissez pas encore cet extra-ordinaire jeu ?

- Essayez-le vite (mais pas trop longtemps) en allant là:
https://archive.org/details/msdos-Rogue_1983.
- ou Parcourez rapidement [https://en.wikipedia.org/wiki/Rogue_\(video_game\)](https://en.wikipedia.org/wiki/Rogue_(video_game)) ou <https://en.wikipedia.org/wiki/NetHack>

Vous savez enfin à quoi jouaient les élèves au début des années 80...



rogue - wikipedia

La suite de ce document vous donne une idée du jeu avec un graphique hyper sophistiqué en caractères **ascii**. Vous l'adapterez aux bibliothèques graphiques que vous connaissez ou préférez.

Les débutants en c++, trouveront une base d'un code qui déplace le personnage en caractère **ascii** dans le repository `git@github.com:ue12-p25/cpp-nethack-rogue.git` à cloner donc puis lire le `readme.md`.

1 À quoi ressemblerait le tout premier jeu Nethack-Rogue

Voici quelques idées sur la manière (approximative) dont fonctionnait **Nethack-Rogue**. Libre à vous de vous en inspirer (ou pas).

Dans un terrain de jeu, constitué des lignes et des colonnes d'un terminal, vous contrôlez votre héro en entrant des caractères au clavier (par exemple des touches pour se déplacer).

Votre *personnage*, le célèbre '@', erre dans le labyrinthe d'un donjon à la recherche d'un Graal (les '+' sont des portes, les '#' les couloirs, le '=' l'escalier menant au niveau supérieur - au début un seul niveau suffit à '@')

```
-----
|. @ . . . | ##+ . . . | #####
|. . . . | # | . . . | # #
|. . . . +### | . . . +### #
|. . . . | | . . . | #
|. . . . | ##+ . . . | #
|. . . . | # ----+--- #####
-----+--- # # #
# # # ---+---
# # # | . . . |
##### | . . = |
-----
```

figure 1: '@' dans la première pièce d'un niveau de 3 pièces

Notons qu'il est plus sympa que le donjon se découvre petit à petit au fur et à mesure du déplacement du personnage (mais dans un premier temps, il peut être plus simple de tout afficher).

Au cours de sa quête, votre personnage rencontre des ennemis qu'il doit combattre vaillamment (par exemple en leur avançant dessus ou par toute autre manière).

Votre personnage s'affaiblit quand il prend des coups (ses points de vie diminuent) mais quand il en sort vainqueur, sa force augmente et il gagne des points de vie (par exemple).

```

-----
|. @ . . . | K# + . . . | #####
|. . . . | # | . . . | # #
|. . . . +### | . . . +### #
|. . . . | | . . . | #
|. . . . | ### + . . . | #
|. . . . | # - - - + - - #####
- - - + - - # # #
# # # - - + - -
# # # | . . . |
##### | . . = |
-----

```

figure 2: '@' ne voit pas encore l'horrible 'K' caché dans un couloir

Votre personnage va trouver des trésors et des objets durant sa quête:

- de l'or '*'
- potions magiques 'j' qu'il peut ramasser, placer dans son sac pour les boire plus tard (parfois au dépend de ses points de vie ou de force mais elles peuvent aussi lui donner des pouvoirs magiques ou augmenter sa force...).
- armes, poignards, dagues, épées '!', arcs et flèches ')' qu'il peut ramasser, placer dans son sac ou prendre en main pour combattre des monstres.
- armures ou casques, parchemins, anneaux aux pouvoirs magique.
- nourriture, eau (parce que votre personnage ne doit pas mourir de faim ni de soif)

```

-----
|. @ . . . | ## + . . ! . | #####
|. . . . | # | . & . . | # #
|. . . . +### | . . . +### #
|. . . . | | . . o . | #
|. . j . . | ## + . . . | #
| * . . . | # - - - + - - #####
- - - + - - # # #
# # # - - + - -
# # # | . . . |
##### | . . = |
-----

```

figure 3: '@' avec une potion, de l'or, un poignard, un anneau, une armure.

Quand il avance sur un objet, le jeu décrit l'objet. Par exemple: "Vous avez trouvé 30 pièces d'or" (celles-ci se rajoutent automatiquement à son porte-monnaie). Ou encore "Vous avez trouvé une potion magique" (qu'il ramasse automatiquement ou non)

Le donjon peut avoir plusieurs niveaux (d'où les escaliers). Il peut avoir des passages secrets, des trappes... Le personnage peut faire des tas d'actions (avancer, monter un escalier '<', le descendre '>', combattre, lire un parchemin, manger, boire, s'armer, enfiler une armure, mettre un casque, allumer sa torche (des pièces peuvent être plongées dans l'obscurité) ...mais nous vous recommandons d'y aller progressivement.

Et votre personnage peut enfin trouver le Graal...

2 Les consignes

1. découpage

Prenez un moment en début de séance pour découper le jeu, décider "qui fait quoi" et comment les "morceaux" s'articulent (s'interfacent).

2. intégration

Réalisez et réunissez tout de suite une version initiale très simple des différents morceaux dans un même programme qui fonctionne.

3. agilité

Puis chacun doit pouvoir proposer des versions de plus en plus évoluées de son "morceau" mais surtout partez d'une version hyper minimaliste et complexifiez-la au fur et à mesure.

4. gestion de versions

Utilisez `git/github` pour réunir vos codes. Ayez toujours une *dernière version* qui fonctionne (pour pouvoir y revenir)

5. finalité

C'est une expérience de programmation en groupe en temps limité avec sujet imposé. L'important est de vous impliquer, de jouer le jeu, de garder votre motivation tout au long de la journée et de vous amuser.

En cas de blocage, de démotivation ou de segfault: vous pouvez appeler l'enseignant de votre salle (ou pas loin).