



v1.0 **Projet CY-'dventure**

CLASSE préING1 • 2022-2023

AUTEUR Romuald GRIGNON

E-MAIL romuald.grignon@cyu.fr

D'APRES UNE IDEE ORIGINALE DE groupe MI5

DESCRIPTION

Ce projet a pour but de développer un programme qui va dérouler une histoire comme dans un "Livre dont vous êtes le héros".

Le principe de ces livres est de proposer une histoire qui possède plusieurs arcs, qui sont déroulés en fonction des choix du lecteur. C'est donc une histoire interactive où les choix du lecteur influent sur la suite des événements. Toutes les variations de l'histoire, les différentes branches possibles, sont déjà écrites, mais en fonction des décisions prises, seule quelques unes seront parcourues. Plusieurs lectures de l'histoire peuvent donc fournir une immersion différente mais également des fins différentes.

[FICHIERS DE L'HISTOIRE]

Votre programme ne devra pas contenir les éléments de l'histoire : le programme est un "programme de lecture" de différents fichiers qui contiennent l'histoire, et les liens entre les différents moments. Nous appellerons chacun des fichiers, une **étape** : l'histoire étant une succession d'étapes, avec potentiellement des décisions/actions à effectuer pour passer d'une étape à une autre.

Par défaut chacun des fichiers de votre histoire contient une seule étape. Le nommage de ces fichiers est laissé libre mais il est conseillé d'utiliser un format simple pour que le programme puisse aller chercher le bon fichier en fonction de l'étape en cours.

Un fichier d'étape est constitué des blocs suivants dans l'ordre (à vous de définir le format exact et les séparateurs entre blocs) :

> Bloc de description :

un bloc avec la description du chapitre : il contient une ou plusieurs lignes de texte qui seront affichées à l'écran afin que le lecteur puisse découvrir l'histoire. Le programme devra s'adapter automatiquement pour afficher la totalité de la description lors du chargement de l'étape courante. Une fois la description de l'étape de l'histoire affichée, le programme va traiter le bloc suivant.

Une évolution intéressante serait d'afficher sur le terminal les caractères les uns après les autres au fur et à mesure pour simuler l'écriture de

l'histoire à l'écran (un peu à l'image de certains jeux d'aventure où l'on voit les dialogues défiler).

> Bloc d'événement :

un événement est un nommage très générique : il peut s'agir d'une modification des statistiques de votre personnage, une modification de son inventaire, ou bien une rencontre avec un ennemi qu'il faut combattre).

Ces événements impliquent que le personnage de votre histoire contienne a minima des caractéristiques et un inventaire. Il peut en être de même des ennemis rencontrés.

Comme un événement peut faire intervenir différents éléments de jeu, c'est à vous de fournir le format de ce bloc (identifiants du type d'événement, données spécifiques pour traitements).

Une fois l'événement terminé, si votre personnage est encore en vie, votre programme va traiter le bloc suivant.

> Bloc de choix

un bloc contenant des choix que le lecteur devra faire. Pour chaque choix, il faudra un texte de description, et un numéro qui sera le numéro de l'étape suivante. Le nombre de choix est libre, entre 0 et autant que nécessaire pour la cohérence de l'histoire. Le programme devra automatiquement s'adapter au nombre de choix.

Votre programme devra automatiquement demander au lecteur son choix, et ira charger en mémoire l'étape correspondante au choix effectué. Votre programme vérifiera que le choix est possible (à vous de voir si vous voulez demander à saisir une lettre A, B, C, D, ... ou de chiffres 1, 2, 3, 4). Les numéros ou lettres des choix à faire devront être rajoutés automatiquement par votre programme : ils ne doivent pas être fournis dans le fichier d'étape. Le bloc de choix ne contient QUE la description du choix et le numéro d'étape suivante associé.

Si le bloc de choix ne contient qu'un seul choix, alors au lieu de demander de choisir l'étape suivante, le programme demandera d'appuyer sur la touche entrée pour passer à la suite (mais on laisse le temps au lecteur de choisir à quel moment il passe à la suite malgré tout).

Si le bloc de choix ne contient aucun choix possible, votre programme sait que l'histoire est donc terminée. Cela sera le cas pour toutes les fins alternatives de votre histoire. Nécessairement, une étape de l'histoire qui n'est pas la fin contient au moins 1 choix possible dans le bloc de choix.

[FICHIERS DES PERSONNAGES]

Comme indiqué, votre personnage, ou tout autre personnage ennemi de l'aventure, possèdent des caractéristiques et des inventaires qui seront clairement définies dans le code à l'aide de structures. Les valeurs de toutes les caractéristiques et inventaires doivent donc être stockées dans des fichiers. Un fichier par personnage/ennemi. Ainsi lors d'un événement de combat, le programme chargera le monstre correspondant, en initialisant les valeurs des structures.

De la même manière, le personnage principal qu'incarne le lecteur, sera chargé en début d'histoire avec ses caractéristiques de départ. Ces caractéristiques pouvant évoluer en fonction de l'histoire. Votre programme devra coder tous les événements possibles qui peuvent arriver dans votre histoire (le personnage a trouvé de l'or qu'il peut stocker dans son inventaire, il a bu une potion de force qui modifie ses dégâts de manière permanente, il combat un monstre et va donc sortir blessé du combat, si il sort vainqueur, ...).

[FICHIERS DE SAUVEGARDE]

A tout moment, en plus du choix de la prochaine étape, le lecteur pourra sauvegarder l'avancement de sa partie. C'est à dire qu'il soit sauvegarder dans un seul fichier l'avancement de son histoire.

Cette sauvegarde pourra être reprise au démarrage du programme via un petit menu de sélection (nouvelle partie / reprendre une partie).

Vous pouvez faire en sorte de sauvegarder plusieurs parties si vous le souhaitez mais à minima au moins UNE sauvegarde doit être possible (une sauvegarde ultérieure écrasant la précédente).

INFORMATIONS GENERALES

Taille de l'équipe

Ce projet est un travail d'équipe. Il est autorisé de se réunir en groupe de 3 personnes.

Démarrage du projet

Vous obtiendrez de plus amples informations quant aux dates précises de rendu, de soutenance, les critères d'évaluation, le contenu du livrable, ..., quand le projet démarrera officiellement.

Dépôt de code

Vous devrez déposer la totalité des fichiers de votre projet sur un dépôt central Git. Il en existe plusieurs disponibles gratuitement sur des sites comme github ou gitlab.

Démonstration

Le jour de la présentation de votre projet, la version finale sur votre dépôt sera celle utilisée, même si vous avez ajouté des fonctionnalités ou corrigé des bugs entre temps. La démonstration se fera sur une machine de l'enseignant chargé de suivre votre groupe. C'est la date de commit sur votre dépôt qui fera foi.

Vous ferez votre démonstration, en fonction des exigences du cahier des charges de votre projet, et vous aurez à modifier légèrement votre application en direct en fonction de la requête de votre chargé de TD.

De plus des questions supplémentaires pourront être posées afin d'évaluer votre connaissance de l'implémentation de votre projet.

FONCTIONNALITES PRINCIPALES DU PROJET

- Le but principal du projet est de fournir une application **fonctionnelle**.
- Tous les **éléments de votre code** seront écrits en langue **anglaise** (structure, types, fonctions, fichiers, ...).
- Votre code sera généreusement **commenté** (langue française autorisée).
- Votre application ne doit **jamais** s'interrompre de manière intempestive (crash), quelle que soit la raison. Toutes les erreurs

doivent être gérées correctement. Il est préférable de d'avoir une application stable avec moins de fonctionnalités qu'une application contenant toutes les exigences du cahier des charges mais qui plante trop souvent. Une application qui crée des fautes de segmentation par exemple sera très pénalisée.

- Votre application devra être **modulée** afin de ne pas avoir l'ensemble du code dans le même fichier.
- Votre **livrable** sera une **URL** d'un dépôt **Git public** envoyée à votre chargé de TD.
- Votre projet doit pouvoir respecter toutes les règles du cahier des charges et fonctionner avec n'importe quels fichiers d'histoire fournis. Si les fichiers ne sont pas cohérents entre eux, et/ou si il possèdent des défaut de structure, le programme doit le détecter et se stopper proprement.
- Les fichiers d'histoire sont stockés dans un sous-répertoire. A vous de définir le format de nommage mais chaque fichier doit contenir le numéro de l'étape.
- Le programme doit charger l'étape 0 au démarrage (sauf dans le cas d'une reprise d'une partie sauvegardée)
- Les fichiers de personnages doivent être stockés dans un autre sous-répertoire. Idem, vous devez définir le nommage utilisé pour les différents fichiers.
- Les fichiers de sauvegarde doivent être stockés dans un troisième sous-répertoire.
- Faites en sorte que les sous-dossiers soient déjà créés lorsque vous compilez le programme avec votre Makefile, pour éviter les problèmes d'accès depuis votre programme.

RESSOURCES UTILES

Github

- www.github.com
- <https://docs.github.com/en/get-started/quickstart/hello-world>

Modifier les couleurs du terminal Linux

- https://en.wikipedia.org/wiki/ANSI_escape_code#Colors
- <http://sdz.tdct.org/sdz/des-couleurs-dans-la-console-linux.html>

Contrôler le curseur du terminal

- [https://en.wikipedia.org/wiki/ANSI_escape_code#CSI_\(Control_Sequence_Introducer\)_sequences](https://en.wikipedia.org/wiki/ANSI_escape_code#CSI_(Control_Sequence_Introducer)_sequences)

Les livres dont VOUS êtes le héros

- https://fr.wikipedia.org/wiki/Un_livre_dont_vous_%C3%AAtes_le_h%C3%A9ros