

Projet C

I) Consignes

- Vous utiliserez le compiler GCC.
- Vous utiliserez la version C99 (rappel de compilation : `gcc -std=c99`).
- Vous écrirez un fichier Makefile pour compiler et nettoyer votre projet à l'aide des commandes `make build` et `make clean`.
- Vous écrirez un fichier `README.md` pour présenter votre projet.
- Les bibliothèques natives sont autorisées (ex: `stdlib`, `stdio`, etc ...). Les autres bibliothèques ne sont pas autorisées. En cas de doute sur une bibliothèque je vous invite fortement à venir me le demander.
- La génération de code à l'aide d'IA(s) (GPT, Mistral, Copilot, Cursor, ...) n'est pas recommandée.
- Vous commenterez votre code au format javadoc.

II) Notation

Vous présenterez votre projet via un oral. L'objectif est de présenter :

- Les choix que vous avez fait (structure, organisation, représentation, ...).
- Les difficultés que vous avez rencontrées (les sources et vos solutions).
- Si avec du recul vous êtes insatisfait de certaines parties de votre code.

Je vous questionnerai sur votre projet pour m'assurer que vous avez bien compris les notions abordées pendant le cours. Je vous encourage donc une nouvelle fois à ne pas copier de code et à ne pas faire appel à des IA(s). Cela pourrait vous porter préjudice.

J'évaluerai votre projet via les critères suivants :

- Votre projet fonctionne. Je souligne qu'il vaut mieux avoir un projet incomplet mais qui fonctionne que l'inverse. Je vous invite également à bien tester votre projet.
- Votre projet implémente les fonctionnalités demandées.
- Votre code est viable (pas de fuite mémoire, pas de segfault, ...).
- Le projet utilise les structures de données appropriées.
- Votre projet est propre et organisé (respect des conventions, ...).
- Votre projet est correctement commenté.
- Le projet est simple d'utilisation et ergonomique.

Projet C

- Bonus :
 - Vous avez optimisé votre solution (temps de calcul, mémoire utilisée).
 - Vous avez choisi le sujet 1.
 - Vous êtes allés plus loin (ajout de fonctionnalités).

III) Contexte

Le SVG est un format graphique qui représente des formes via des vecteurs pour assurer qu'il n'y a pas de perte de résolution. Un SVG est un plan en 2 dimensions sur lequel on peut venir y dessiner des formes ou des trajectoires. Un SVG est composé de deux éléments :

- Un « viewport » : Il s'agit d'une fenêtre sur le plan en deux dimensions. Seuls les éléments visibles dans cette fenêtre sont dessinés.
- D'une liste d'éléments graphiques qui peuvent être stylisés.

IV) Sujet

Vous êtes chargé de développer une interface par ligne de commande pour créer, éditer, sauver, charger et visualiser des structures graphiques similaires au format SVG. Je précise ici que vous êtes libre d'adapter le format SVG tant que vous respectez le principe. Seules les formes suivantes sont nécessaires pour ce projet :

- Les cercles
- Les ellipses
- Les carrés
- Les rectangles
- Les lignes
- Les enchainements de lignes
- Les polygones
- Les trajectoires ("chemins")
- Les groupes

Vous trouverez à ce lien

<https://developer.mozilla.org/fr/docs/Web/SVG/Reference/Element/svg> les détails de chaque forme.

Projet C

Je souhaite pouvoir styliser chaque forme, en particulier modifier :

- La couleur de trait
- La couleur de fond
- Les déplacer en X et en Y (entier).
- Les pivoter selon un angle (entier).
- Les inverser selon l'axe X ou Y.

A vos ordinateurs !