

Projet Informatique – Sections Electricité et Microtechnique

Printemps 2019 : [Archipelago](#) © R. Boulic

Rendu3 (dimanche 24 mai 23h59)

Objectif de ce document : comme pour le document des rendus précédent, ce document identifie des **ACTIONS** à considérer pour réaliser le rendu de manière rigoureuse. Le présent document suit l'architecture minimale du projet (donnée Fig10 et **Fig 11 b1**). A nouveau, on s'appuiera sur la série [méthodes de développement de projet](#).

1. Buts du rendu3 (Donnée section 7 rendu3)

Vue d'ensemble : Edition, performances, rapport.

- Action d'ajout et de retrait de nœud ou de lien
- Déplacement du nœud courant
- Gestion de changement de taille de fenêtre, de facteur de zoom.

Ce rendu sera testé en lançant des scénarios de test de complexité croissante pour estimer les performances de votre approche.

1.1 Syntaxes du lancement de l'exécutable :

Les deux syntaxes des rendus précédents, de même que le comportement en cas d'erreur à la lecture (sur fichier e01.txt), doivent continuer à fonctionner.

1.2 Architecture (Fig 11 de la donnée) :

Même architecture que le rendu2.

L'interface étant déjà en place la section suivante propose certaines actions par fonctionnalité à réaliser.

2 Actions proposées :

2.1 ACTION : Transformation de coordonnées de la souris de l'espace GTmm vers l'espace du Modèle

Avant de se lancer dans l'édition elle-même il faut s'assurer que les coordonnées de la souris sont converties correctement dans l'espace du Modèle. Le cours sur la gestion d'événement « souris/clavier » décrit comment le faire en prenant soin de prendre en compte l'éventuel décalage de la surface de dessin (slide5) avant d'appliquer les formules vers l'espace du Modèle (slide 9).

Des ressources sont disponibles dans la **série7 niveau0 exercice2**. Dans cette phase de mise au point, faites aussi afficher dans le terminal les coordonnées brutes, celle dans l'espace de dessin et celle de l'espace du Modèle. Un petit programme de test qui affiche un cercle centré sur le point cliqué est suffisant à ce stade.

2.2 ACTION : Transformation du cadrage pour éviter les distorsions du dessin

Puisque vous venez de vous frotter à la conversion de coordonnées dans le cas simple, autant traiter le cas compliqué dans la foulée... C'est le cas où on change la taille de la fenêtre pendant l'exécution. Cela produit un appel automatique de `on_draw`. Il est donc important de récupérer dès le début de cette méthode les nouvelles largeur et hauteur puis de faire un ajustement du cadrage [`Xmin,Xmax`] et [`Ymin,Ymax`] pour éviter toute distorsion (slide 16 du cours sur le dessin avec GTKmm). La méthode à suivre a été traitée dans l'**exercice 3c de la série6**.

Ré-utilisez le petit programme qui affiche des cercles, changez la taille de la fenêtre et on doit toujours voir des cercles. De plus si vous avez fait suivre l'information du nouveau cadrage, la transformation de coordonnée vers le Modèle devrait aussi être correcte : un cercle doit apparaître exactement centré à l'endroit où on a cliqué.

2.3 ACTION : Transformation du cadrage avec zoom-in, oom-out et zoom-reset

Dans l'état initial le facteur de zoom vaut 1 pour le cadrage indiqué par la donnée [`-dim_max, dim_max`] avec la fenêtre de taille initiale (sinon voir 2.2). Calculer le nouveau cadrage en gardant le centre fixe dans l'espace de dessin. Vérifier en faisant afficher les coordonnées converties de la souris dans le nouveau cadrage.

Attention au Zoom_reset : il faut prendre en compte que la taille de la fenêtre a peut-être changé en largeur/hauteur. Il faut mettre à jour un cadrage de « reset » chaque fois qu'on change la taille de la fenêtre.

2.4 ACTION : usage des coordonnées dans l'espace du Modèle pour sélectionner/détruire le nœud courant

Un clic gauche à l'intérieur d'un nœud en fait le nœud courant qui est visualisé en rouge. C'est un bon moyen pour vérifier que la transformation des coordonnées est correcte et que votre fonction « point dans cercle » de tools est correcte aussi.

Vérifier que le nœud choisi est affichée en rouge, qu'il change si on clique sur un autre nœud ou qu'il n'y a plus de nœud dessiné en rouge si on clique à l'extérieur de tout nœud.

Un second clic gauche sur le même nœud doit détruire ce nœud. Vérifier que vous pouvez détruire tous les nœuds de cette manière. Commencer par tester une ville avec aucun lien entre les nœuds. Ensuite utilisez des fichiers de test avec des configurations de liens plus complexes.

2.5 ACTION : usage des coordonnées dans l'espace du Modèle pour créer/déplacer/modifier un nœud

La création d'un nouveau nœud implique un test de non-superposition vis-à-vis des *autres* nœuds qui prend en compte la marge de sécurité `dist_min`. Une détection de nœud trop proche empêche la création d'un nouveau nœud. Créer deux nœuds aussi proche que possible avec cette marge `dist_min` puis sauvegarder le fichier et relisez-le ; il ne devrait y avoir aucun problème car la lecture ne prend pas en compte la marge de sécurité (2.1.3).

Même principe de marge de sécurité vis-à-vis des autres nœuds quand on veut déplacer (bouton droit) ou agrandir (bouton gauche) le nœud courant.

Pour agrandir, il faut cliquer en dehors du nœud courant sinon on doit le détruire (6.1.2). De plus il faut rester dans l'intervalle de rayon autorisé.

2.6 ACTION : usage des coordonnées dans l'espace du Modèle pour l'édition des liens

Création : ne pas oublier de prendre en compte la marge de sécurité (2.1.3) pour ré-utiliser votre fonction de non-superposition de lien avec un nœud *différent du nœud de départ et du nœud d'arrivé*.

Destruction : commencer par une ville de 2 nœuds avec un seul lien, poursuivre les tests sur des configurations de plus en plus riches.

2.7 ACTION : affichage de shortest path pour le nœud courant

Votre batterie de tests doit comprendre les configurations pour lesquelles il n'y a pas de shortest path par manque de nœud ou d'accès impossible. Une autre famille de test doit inclure des nœuds transports en séquence pour offrir une alternative plus rapide que le chemin visuellement le plus court (normalement le rendu2 résout déjà cette question ; on en a ici une vérification visuelle).

3 Evaluation du rendu3 :

En plus d'évaluer les critères sur la qualité du code (ne pas oublier de corriger tout warning obtenu au feedback des rendus précédents), nous effectuerons une évaluation manuelle de votre programme comme suit :

- construction d'une ville (minimale) depuis zéro à l'aide de l'interface graphique : ajout/suppression de nœuds des 3 types avec ajout des liens. Les conditions de non-superposition seront testées (section 2.1.3) pour les nœuds et pour les liens.

- sauvegarde et relecture de la ville sauvegardée.

- ouverture d'un fichier de ville simple et destruction de tous les éléments à l'aide des actions de la souris (sélection multiple du bouton gauche)

- ouverture d'un fichier plus gros pour tester la visualisation des plus courts chemins.

- changement de la taille de la fenêtre produisant un changement du rapport largeur/hauteur de la surface de dessin ; l'affichage ne doit pas avoir de distorsion et la conversion des coordonnées de la souris doit rester correcte : un nœud sera créé et doit apparaître à l'endroit où on a cliqué.

4. Rapport final

Entre 2 pages (ma préférence) et 3 pages maximum, avec une police 11 au minimum et 14 au maximum, interligne simple (SVP : ne gaspillez pas de papier avec des pages de titre et de table des matières).

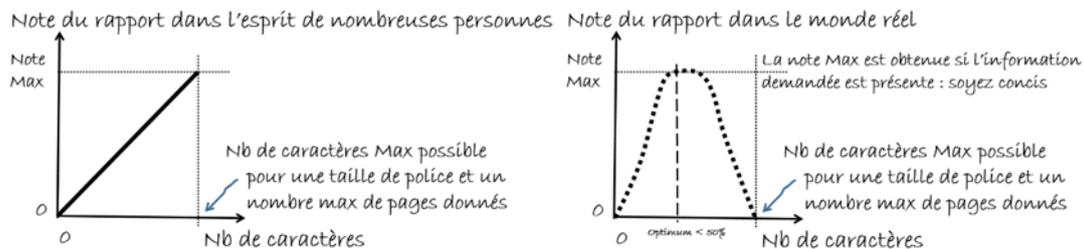


Figure Rapport : un mauvais rapport dilue l'information utile jusqu'à atteindre le nombre maximum de caractères possibles du fait d'une croyance scolaire antérieure (image de gauche) ; un bon rapport est celui qui fournit les informations demandées avec concision (image de droite)

Le rapport est écrit en français ou en anglais ; une orthographe ou une grammaire défailante peut induire les correcteurs en erreur. Le rapport contient :

Architecture logicielle et description de l'implémentation:

- Dessin de l'architecture logicielle choisie Fig11 b1 ou 11 b2, ou une variante accepté par l'enseignant.
- Dessin de la hiérarchie de classes si vous en avez créé une (optionnel).
- Décrire et justifier la répartition des tâches entre les modules du Modèle. Qui calcule quoi ?
- Préciser la structuration des données pour représenter le graphe de la ville (dessin bienvenu si le choix est différent des propositions du cours).
- Comment avez-vous mis en œuvre l'algorithme de Dijkstra (en comparaison du pseudocode fourni dans la donnée). Quelle modification au pseudocode du cours avez-vous fait pour trouver le noeud transport le plus proche d'un nœud logement ?
- Ajouter et commenter les images (recadrée sur la ville) montrant le dessin du plus court chemin pour 2 nœuds différents d'une ville de votre choix mais ayant au moins une douzaine de nœuds avec les 3 types de nœud.

Méthodologie et conclusion : comment avez-vous organisé votre travail à plusieurs, indiquer la personne responsable de chaque module et comment vous avez organisé le travail au sein du groupe (par quels modules avez-vous commencé, comment les avez-vous testés, comment le feriez-vous maintenant avec le recul.

Quel était le bug le plus fréquent, pourquoi ? Quel est celui qui vous a posé le plus de problème et comment a-t-il été résolu, ...). Pour conclure fournissez une brève auto-évaluation de votre travail et de l'environnement mis à votre disposition (points forts, points faibles, améliorations possibles).

Le rapport final doit être inclus dans le fichier archive du rendu final (**en format pdf**).

5. Forme du rendu3

Pour chaque rendu **UN SEUL membre d'un groupe** (noté **SCIPER1** ci-dessous) doit télécharger un fichier **zip** sur moodle (pas d'email). **Le non-respect de cette consigne sera pénalisé de plusieurs points.**

Le nom de ce fichier **zip** a la forme : **SCIPER1 SCIPER2.zip**

Exemple pour le groupe Wagnières/Zimmer: **301438_301900.zip**

Compléter le fichier fourni **mysciper.txt** en remplaçant 111111 par le numéro SCIPER de la personne qui télécharge le fichier archive et 222222 par le numéro SCIPER du second membre du groupe¹.

Le fichier archive du rendu3 doit contenir (**AUCUN répertoire**) :

- Fichier texte édité **mysciper.txt**
- Votre fichier **Makefile** produisant un exécutable **projet**
- votre code source (.cc et .h)
- le fichier **pdf** du rapport

*On doit pouvoir produire l'exécutable **projet** à partir du Makefile après décompression du contenu du fichier zip. La commande **make** ne doit faire AUCUN déplacement de fichier ; tout reste dans l'unique répertoire créé par la décompression du fichier archive.*

Auto-vérification : Après avoir téléversé le fichier zip de votre rendu sur moodle (upload), récupérez-le (download), décompressez-le et assurez-vous que la commande **make** produit bien l'exécutable et que celui-ci fonctionne correctement.

Exécution sur la VM: votre projet sera évalué sur la VM du cours (c'est la même qu'au premier semestre).

¹ L'unique groupe de 3 personnes rajoute le 3ieme SCIPER sur une troisième ligne dans mysciper.txt