

Projet Informatique – Sections Electricité et Microtechnique

Printemps 2023 : *Mission propre en Ordre* © R. Boulic & collaborators

Rendu3 (21 mai 23h59)

Objectif de ce document : Ce document utilise l'approche introduite avec la série théorique sur les [méthodes de développement de projet](#) qu'il est important d'avoir faite avant d'aller plus loin.

En plus de préciser ce qui doit être fait, ce document identifie des **ACTIONS** à considérer pour réaliser le rendu de manière rigoureuse. Ces **ACTIONS** sont équivalentes à celles indiquées pour le projet d'automne ; elles ne sont pas notées, elles servent à vous organiser. Vous pouvez adopter une approche différente du moment que vous respectez l'architecture minimale du projet (donnée Fig 9b).

1. Buts du rendu3 : dialogue avec l'interface graphique et simulation

Votre approche peut évoluer entre ce que vous avez décrit pour le rendu2 en matière de structuration des données et ce que vous mettez en œuvre finalement du moment que vous respectez les responsabilités des différents modules (Fig ci-contre).

Lancement et comportement attendu:

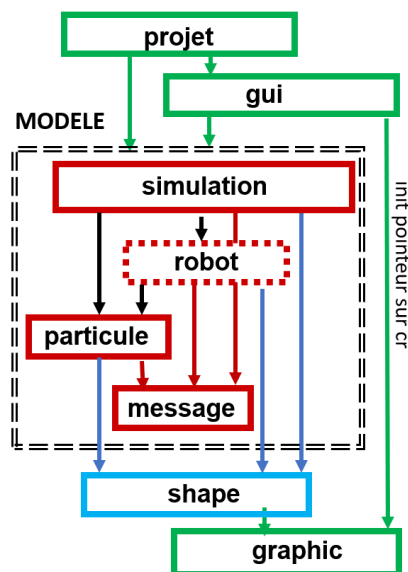
Le programme sera lancé soit en indiquant un nom de fichier à ouvrir (rendu2) :

```
./projet test1.txt
```

Soit sans aucun argument sur la ligne de commande :

```
./projet
```

Dans ce second cas, l'interface est créée et le programme attend qu'on lui demande d'ouvrir un fichier avec le bouton Open.



Donnée Fig 9b

. 1.1 Rapport final (MAX 2 pages):

Le Rapport ne contient PAS de page de titre, ni de table des matières. Il est écrit avec une police 11 au minimum et 14 au maximum, interligne simple. Le rapport est écrit en français ou en anglais ; une orthographe ou une grammaire défailante peut induire les correcteurs en erreur. Le Rapport ne duplique pas le précédent. Il contient :

- **la description de votre approche** pour les algorithmes de la section 2.5 de la donnée générale.
 - 2.5.1 : décision de création d'un nouveau robot
 - 2.5.2 : décision de mise à jour du but des robots réparateurs
 - 2.5.3 : décision de mise à jour de la particule cible des robots neutraliseurs

Captures d'écran (conserver seulement la partie utile) de plusieurs étapes de votre simulation pour le fichier de test t45.txt lorsqu'il y a plusieurs robots neutraliseurs en action. La légende doit expliquer les mouvements des robots d'une capture d'écran à la suivante. Indiquer de nombre de mises à jour entre chaque capture. Fournir une autre série de captures sur un fichier de test de votre choix si vous n'y arrivez pas avec t45.

- **Méthodologie et conclusion** : comment avez-vous organisé votre travail à plusieurs, avec quels outils ? (git?, VSCode ? etc...) ; indiquer la repartition des contributions par module et comment vous avez organisé le travail au sein du groupe (par quels modules avez-vous commencé, comment les avez-vous testés). Indiquer la proportion de travail simultané en groupe (c'est à dire côte à côte ou en-ligne sur le même code) par rapport au travail indépendant (chacun de son côté). Avec le recul, est-ce que vous modifieriez cette proportion?
 - 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).

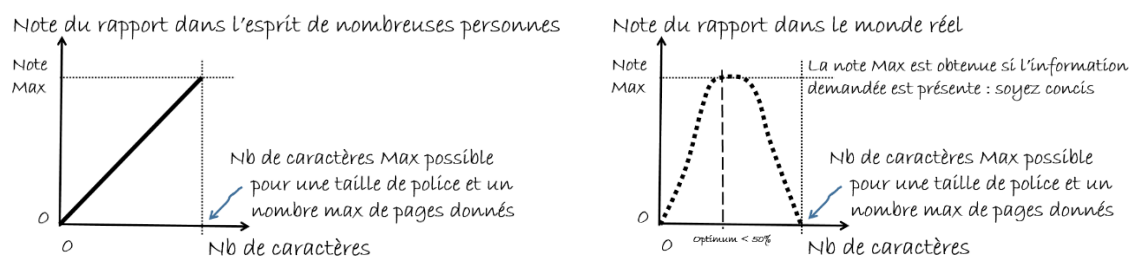


Figure 1 : un mauvais rapport dilue l'information utile jusqu'à atteindre le nombre maximum de caractères possibles sur la page (fig gauche) : en fait ce type de rapport sera pénalisé parce qu'il est peu lisible ; un bon rapport est celui qui fournit les informations demandées avec concision avec une mise en page aérée et lisible (fig droite)

1.2 Evaluation du rendu3 : en plus d'évaluer les critères sur la qualité du code (ne pas oublier de corriger tout warning obtenu aux rendus précédents), nous effectuerons une évaluation manuelle de votre programme comme suit :

- lancement avec les 2 syntaxes : avec et sans nom de fichier sur la ligne de commande.
- évaluation de la mise à jour de la simulation dans différents scénarios pour faire apparaître les comportements demandés. La majorité des scénarios sera constitué des cas très simples discutés dans la section suivante. Seuls les derniers scénarios auront une complexité plus grande pour l'évaluation des performances.

2. Organisation du travail

Nous recommandons de tester votre simulation en mode Step avec l'affichage graphique de chaque état successif de la simulation car c'est un puissant outil de mise au point. Utilisez les fichiers de test fournis.

N'hésitez pas à compléter avec du code de scaffolding qui affiche la valeurs des variables importantes dans le terminal pour compléter vos outils de mise au point. D'ailleurs, en cas de bug, l'affichage dans le terminal devrait être redirigé vers un fichier de texte (cf cours redirection de la sortie) pour pouvoir l'ouvrir avec un éditeur de texte après le crash du programme et analyser l'évolution des valeurs de vos variables.

fichier	Mettre valeur de desintegration_rate à :	But du test: vérifier...
t28.txt	0	<ul style="list-style-type: none"> Le robot neutraliseur de type0 se dirige vers la particule qu'il atteint en 2 mises à jour La particule est supprimée et le robot revient vers le robot spatial Fin de la simulation lorsque le robot est revenu dans la robot spatial
t29.txt	0	<ul style="list-style-type: none"> Le robot neutraliseur de type0 tourne d'abord sur place avant de se diriger vers la particule qu'il atteint en 2 mises à jour La particule est supprimée et le robot revient vers le robot spatial Fin de la simulation lorsque le robot est revenu dans la robot spatial
t30.txt	0	<ul style="list-style-type: none"> Le robot neutraliseur de type0 tourne d'abord sur place avant de se diriger vers la particule qu'il atteint en 5 mises à jour puis il tourne sur place pour s'aligner sur la normale à la particule La particule est supprimée et le robot revient vers le robot spatial Fin de la simulation lorsque le robot est revenu dans la robot spatial
t31.txt	0	<ul style="list-style-type: none"> Le robot neutraliseur de type1 tourne d'abord sur place avant de se diriger vers la particule, en restant en dehors de la zone à risque, puis il tourne sur place pour s'aligner sur la normale à la particule, et enfin il avance vers la particule. La particule est supprimée et le robot revient vers le robot spatial Fin de la simulation lorsque le robot est revenu dans la robot spatial
t32.txt	0	<ul style="list-style-type: none"> Le robot neutraliseur de type2 avance en tournant et finit de s'aligner sur la normale à la particule quand il est en collision avec celle-ci. La particule est supprimée et le robot revient vers le robot spatial Fin de la simulation lorsque le robot est revenu dans la robot spatial
t33.txt	0	<ul style="list-style-type: none"> La particule la plus grande doit être choisie comme cible même si elle est plus éloignée
t34.txt	0	<ul style="list-style-type: none"> La particule la plus grande doit être choisie comme cible par le robot le plus proche
t35.txt	0	<ul style="list-style-type: none"> La particule la plus grande doit être choisie comme cible par le robot le plus proche ; (inversion de l'ordre des robots dans le fichier par rapport à t34)
t36.txt	0	<ul style="list-style-type: none"> La particule la plus grande doit être choisie comme cible par le robot le plus proche ; dans ce cas la différence d'orientation impacte le choix du robot le plus proche
t37.txt	0	<ul style="list-style-type: none"> Une particule se trouve sur le chemin de la particule la plus grande ; elle doit être détruite
t38.txt	0	<ul style="list-style-type: none"> Un robot neutraliseur en panne est détruit après 5 mises à jour car cela correspond à la durée max de 600 depuis qu'il est tombé en panne
t39.txt	0	<ul style="list-style-type: none"> le robot réparateur se dirige vers le robot neutraliseur en panne le plus proche
t40.txt	0	<ul style="list-style-type: none"> le robot réparateur se dirige vers le robot neutraliseur en panne le plus proche MAIS seulement s'il reste suffisamment de temps au robot réparateur pour l'atteindre ; ici il se dirige donc vers l'autre robot neutraliseur en panne
t41.txt	0	<ul style="list-style-type: none"> le robot réparateur rentre dans le robot spatial car aucun robot neutraliseur n'est en panne
t42.txt	0	<ul style="list-style-type: none"> le robot spatial decide de créer un nouveau robot neutraliseur quand nb_update est un multiple de modulo_update. Une seule particule ; un seul robot en reserve.
t43.txt	0	<ul style="list-style-type: none"> le robot spatial decide de créer un nouveau robot neutraliseur quand nb_update est un multiple de modulo_update ; context avec 4 particules et 3 robots en reserve. Plusieurs robots neutraliseurs doivent être créés.
t44.txt	0.002	<ul style="list-style-type: none"> contexte general pour évaluer la robustesse et les performances du programme
t45.txt	0.002	<ul style="list-style-type: none"> autre contexte general pour évaluer la robustesse et les performances du programme

3. Forme du rendu3

Convention de style : il est demandé de respecter les conventions de programmation du cours.

Documentation : l'entête de vos fichiers source doit indiquer le nom du fichier et les noms des membres du groupe avec, pour les fichiers .cc, une estimation du pourcentage de contribution de chaque membre du groupe au code de ce fichier.

Rendu : pour chaque rendu UN SEUL membre d'un groupe (noté SCIPER1 ci-dessous) doit téléverser 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

Compléter le fichier fourni mysciper.txt en remplaçant 11111 par le numéro SCIPER de la personne qui télécharge le fichier archive et 22222 par le numéro SCIPER du second membre du groupe.

Le fichier archive du rendu3 doit contenir (aucun répertoire) :

- Rapport final (fichier pdf)
- Fichier texte édité **mysciper.txt**
- Votre fichier **Makefile** produisant un exécutable **projet**
- Tout le code source (.cc et .h) nécessaire pour produire l'exécutable.

On doit obtenir l'exécutable projet en lançant la commande make après décompression du fichier zip.

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 à distance.

Backup : Il y a un backup automatique sur votre compte myNAS.