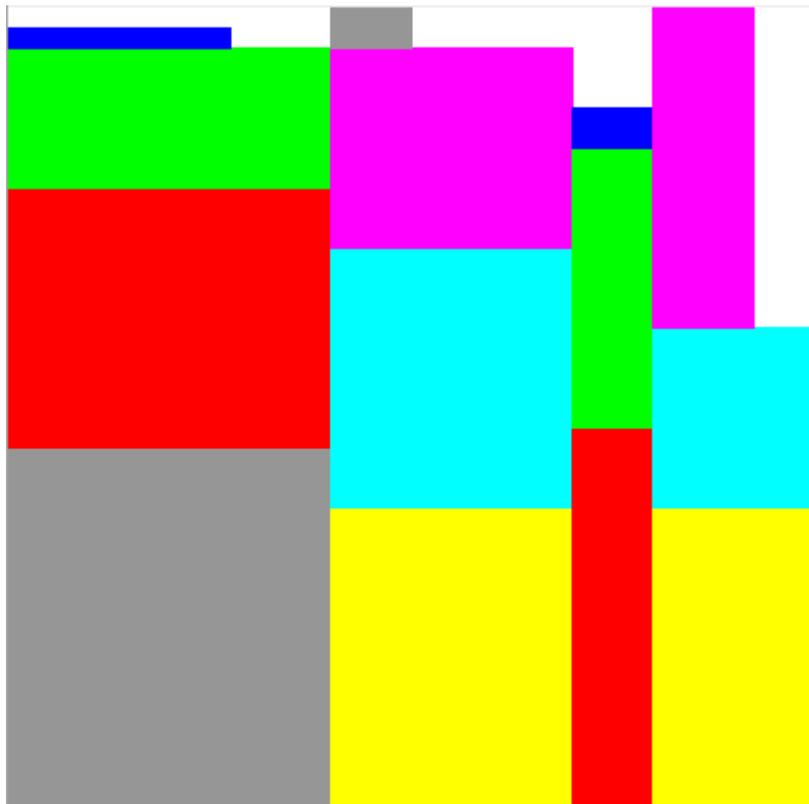


Rapport de projet
L2 Informatique, TP Algorithmique
Packing projet 2d 2014



Binôme :

Masliah Yann

Baghafor Raphael

Rendu le 4 avril 2014

Le sujet du projet

Objectif :

Remplir un conteneur C de largeur L et de hauteur H avec une liste d'objets rectangulaire de largeur L_n et de hauteur H_n .

Problèmes :

Remplir au mieux le conteneur avec les objets fournis.

Les objets doivent avoir leurs bords parallèles aux bords du conteneur.

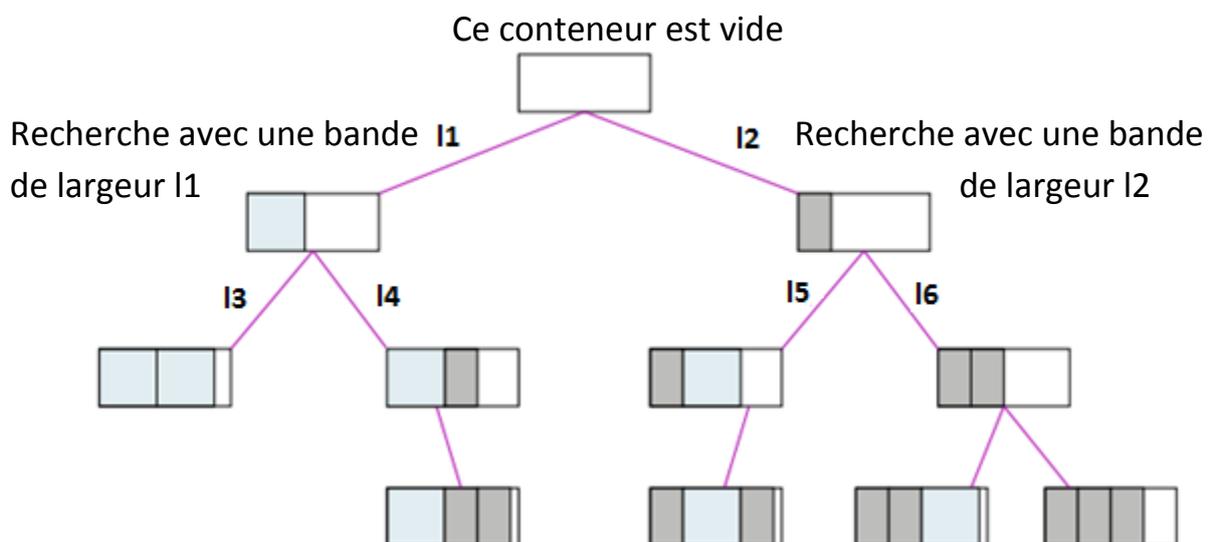
Le meilleur remplissage est calculé suivant la surface totale utilisée par les objets dans le conteneur.

Approche :

Lors de la recherche d'une solution à ce projet, il a été décidé de découper le conteneur en plusieurs bandes pour réduire le coup du programme et donc augmenter sa rapidité d'exécution.

Le coût du programme étant toujours élevé, un certain nombre de largeur de bande sera tester pour le remplissage du conteneur.

Ci-dessous un exemple pour 2 largeurs de bande



Chaque nouvelle bande essaie de remplir le maximum de surface restante dans le conteneur (surface blanche).

Les largeurs valides sont choisies suivant la largeur d'un objet disponible.

Implémentation

3 exécutables sont créés :

Packing : récupère les valeurs du conteneur et des objets dans un fichier txt puis remplit au mieux le conteneur

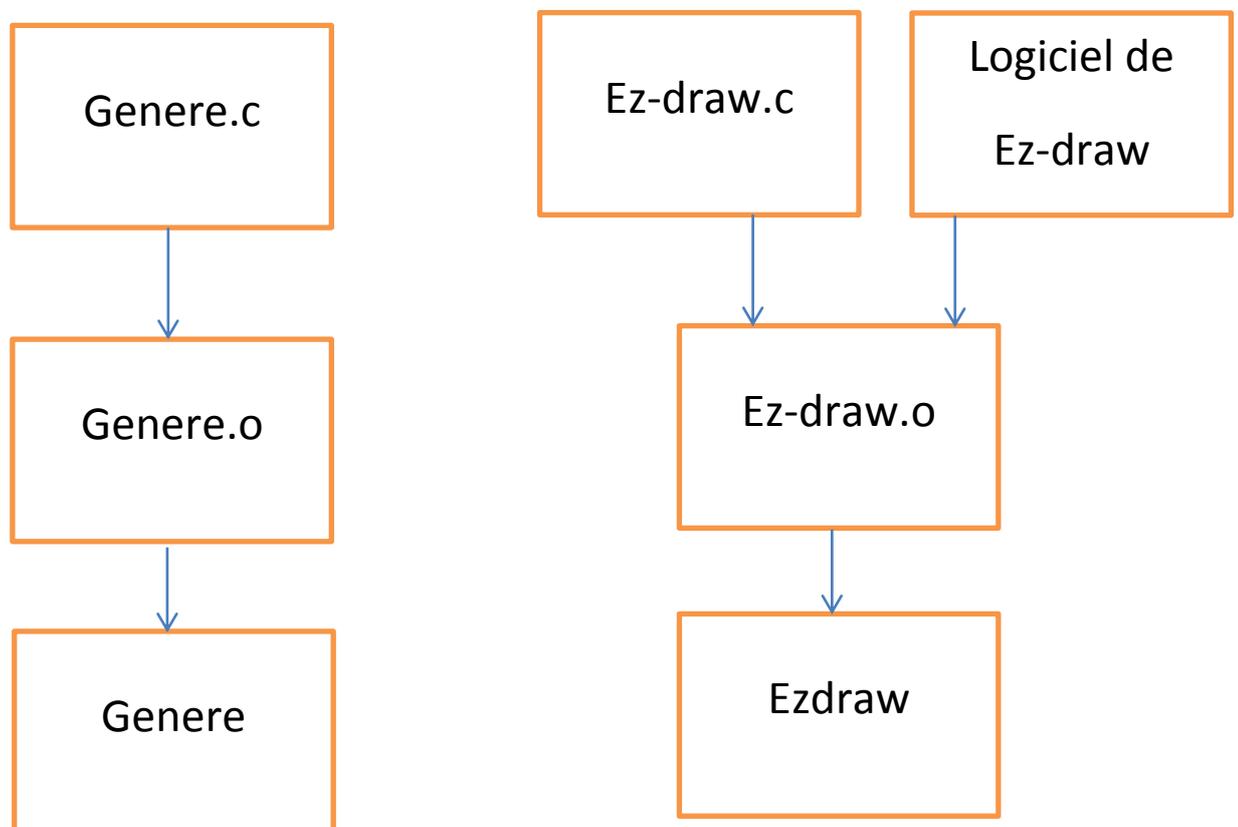
Genere : crée un problème dans un fichier Genere.txt

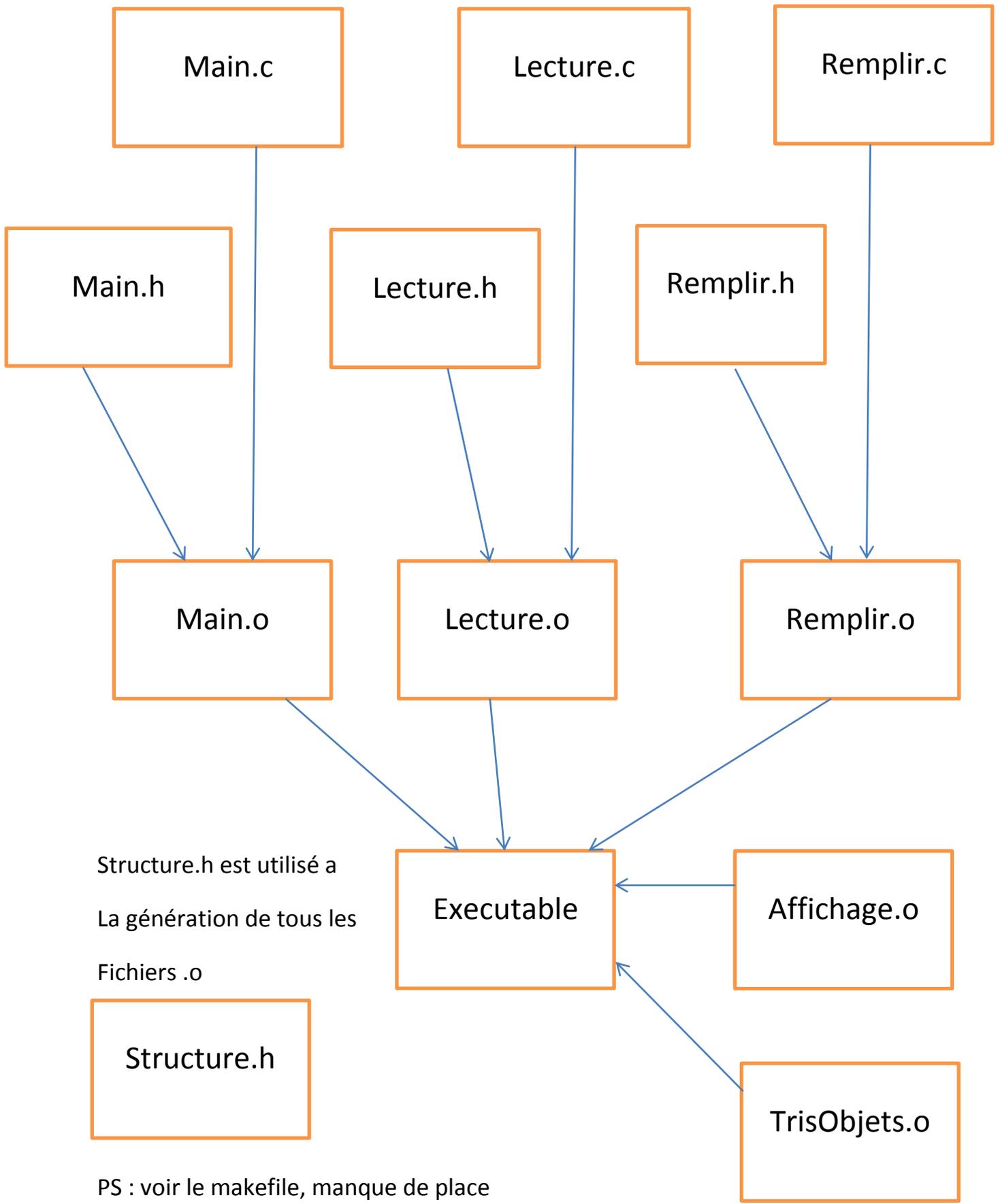
Ezdraw : affiche le résultat récupéré dans un fichier Resultat.txt

L'utilisateur a besoin de fournir le nombre d'objet a créé par l'exécutable Genere puis il crée un conteneur qui peut contenir au moins 1/3 des objets.

De cette manière le remplissage optimal sera plus facilement incomplet.

L'exécutable packing récupère toutes les informations que l'exécutable Genere a écrites dans le fichier Genere.txt et les rajoute dans plusieurs variables globales.





Les valeurs des objets sont contenues dans une structure qui contient 2 champs, une largeur et une hauteur. De cette structure a été créé un tableau qui contient toute les valeurs des objets.

La liste d'objet est triée suivant la largeur puis un sous triage est fait suivant la hauteur, ce qui permet lors du remplissage des bandes une répétition d'objet identiques.

A partir de ce moment la liste d'objet n'es plus modifier, tout se fait dorénavant sur un tableau qui se parcourt suivant les identifiants des objets, contient une valeur 0 si l'objet est déjà utilisée et une valeur 1 si il est encore disponible.

La fonction de remplissage du conteneur est récursive, elle dispose de 6 variables en entrées.

ActualLZone : la longueur actuelle restante dans le conteneur

ObjetRest : le tableau binaire qui indique les objets utilisés

NbRecherche : le nombre de recherches différentes à faire pour chaque longueur restante du conteneur

ActualSurfaceUsed : La surface utilisée par les bandes précédentes.

P : la profondeur de la récursivité, soit le nombre de bande.

Save : contient les largeurs de bande précédentes à la recherche actuelle.

La recherche de la largeur de la bande est choisie suivant la largeur des objets disponibles. Si aucune largeur n'est trouvée, elle fait un second essai avec les objets permutés. (largeur \leftrightarrow hauteur)

Le remplissage de la bande s'effectue en 2 temps :

Le premier remplissage s'effectue seulement avec les objets qui ont une largeur égale a la largeur de la bande, si la hauteur maximal n'es pas atteinte un second remplissage est effectuée avec les objets qui ont une largeur inferieur a la largeur de la bande.

Dans les 2 remplissages la fonction teste si elle peut rentrer les objets en permutant la hauteur et la largeur de l'objet.

La fonction retourne le nombre d'objet utilisé pour le calcul de la surface utilisée et la restitution des objets pour le remplissage de la prochaine bande à cette profondeur.

La fin de la récursivité est provoquée quand aucune largeur de bande n'est disponible.

Une comparaison entre le meilleur remplissage du conteneur (variable globale) et le remplissage du conteneur actuel est effectuée. Si le remplissage actuel est meilleur, elle est sauvegarder a la place de la valeur actuelle et toute les largeurs de bande et le nombre de bandes sont enregistrer dans une autre variable globale qui est BestP .

BestP est un tableau qui contient dans son premier élément le nombre de bande puis les largeurs des bandes.

Lorsque le meilleur remplissage a été effectué, une fonction reconstruit le remplissage pour l'afficher dans le terminal et l'écrire dans un fichier txt pour l'affichage avec une interface graphique.

Améliorations/effets

Plusieurs améliorations ont été faite lors du choix de la largeur des bandes et le remplissage des bandes, grâce aux pivotassions des objets et aux plusieurs comparaison quand les largeurs sont égales ou inferieur.

Apres plusieurs comparaisons entre les 2 modes l'effet est notable, surtout du au faite que le remplissage est glouton, mais en contrepartie les fonctions de remplissage de bande et de choix de largeur de bande ont doublé de volume.

Une autre amélioration a été faite au test de fin de remplissage, si la hauteur maximum a été atteinte la recherche s'arrête, a permis de réduire de 1/3 le temps d'exécution suivant les problèmes.

Problèmes

Le problème le plus dure à trouver a été lors de la création des nouvelles bandes, il créait dans tous les cas le nombre de recherche demandé ce qui me faisait une profondeur infini car il trouver toujours une solution pour la première recherche et aucune valeur pour les autres recherche. Il a suffi que je rajoute un test sur la largeur de la bande avant le remplissage, si elle est égale à 0 je sors de la boucle for qui fait la récursivité.

Expérimentations

La pluparts des expérimentations ont été réalisée avec ez-draw qui permet l'affichage du conteneur.

Suivants certains problèmes qui peuvent contenir un très grand nombre d'objets dans le conteneur (100+), une difficulté de recherche de 4 met déjà quelques secondes à se réaliser.

Il arrive souvent qu'il me reste un objet de 1pixel sur 1 pixel dans ma liste d'objet mais qu'il ne soit utilisé dans aucune bande, d'après moi une des améliorations possible à faire est

- soit de considérer tous les espaces restants comme de nouvelles bandes et essayer de les remplir
- soit d'ajouter des objets à coter de l'objet courant jusqu'à atteindre la largeur de la bande et la hauteur de l'objet courant.

Je n'ai pas eu le temps de réaliser la seconde méthode.