



# Piscine C++ - Rush 2

Pere Noel

Lionel "Iopi" Auroux [auroux\\_l@epitech.eu](mailto:auroux_l@epitech.eu)

*Abstract: Ce document est le sujet du Rush 2*

# Table des matières

I	Les elfes du Père Noël	2
II	Petit Poney et Nounours	3
III	Une histoire de boîte (box tale)	4
IV	Tapis qui roule n'amasse pas mousse	6
V	Travail d'elfe (ou pourquoi je ne ferais de liste de Noël)	8
VI	Tiens v'la la boîte vieux schnock	9
VII	Le Père Noël est-il quantique ?*	10
VIII	The Warp Machine ou un degré de plus avant la démence.	11

# Chapitre I

## Les elfes du Père Noël



Le père Noël a fait une demande officielle à Epitech pour informatiser sa chaîne d'emballage de cadeaux. Il en a marre de ces tafioles d'elfes et voudrait si possible robotiser tout cela. Il vous demande de lui concevoir un simulateur de chaîne d'emballage.



Attention : Ce sujet est volontairement opaque. Il vous est donc demandé de réorganiser les choses logiquement et de déduire les sous-entendus afin de répondre à l'ensemble des exigences. Une lecture INTÉGRALE du sujet est nécessaire pour comprendre toutes les interactions. Non, les koalas ne fument pas d'eucalyptus avant d'écrire leur sujet !



Vous DEVEZ utiliser le dépôt SVN fourni par le Koalab pour votre rendu. L'adresse du dépôt vous sera communiquée par mail automatique à son ouverture.

## Chapitre II

# Petit Poney et Nounours



Il vous fournit une liste décrivant le métier de l'elfe sur la chaîne d'emballage de cadeaux. Votre chef de projet a préféré prendre les choses en main et a traité pour vous l'analyse fonctionnelle et certains aspects de conception. Il vous fournit une liste d'indications et de contraintes pour l'implémentation des différentes parties du simulateur.

Puis il vous indique :

- Un `Nounours` est un `Jouet` .
- Il existe aussi un `Jouet PetitPoney` .
- `Nounours` crie "gra hu" quand on le prend (méthode `estPris` ).
- Le `PetitPoney` crie "yo man" quand on le prend (méthode `estPris` ).
- Un `Jouet` est un `Objet` .

Il vous demande d'écrire les classes `Jouet` , `Nounours` , `PetitPoney`...  
Il veut que vous implémentiez la fonction:

```
1 Objet **MesTestUnitaires();
```

Dans celle-ci, il veut que vous testiez la construction d'un objet petit poney avec un intitulé "poney gay", et un nounours "bisounours". Vous devez les retourner respectivement dans un tableau de 2 éléments.

Vous avez le droit d'utiliser les iostreams pour écrire sur la sortie standard. Vous revenez à la ligne après chaque écriture.

## Chapitre III

# Une histoire de boite (box tale)



Il faut maintenant être capable de gérer l’emballage des cadeaux:

- Un `Carton` est un `Emballage`.
- Un `PapierCadeau` est aussi un `Emballage`.
- L’elfe emballe via la méthode `emballeMoiCa` de `Emballage`.
- L’elfe doit pouvoir mettre un `Objet` (et un seul) dans un `Carton` en l’emballant dans ce carton si et seulement si le `Carton` est ouvert.
- Un `Carton` fermé ne peut emballer.
- Un `PapierCadeau` n’a pas besoin d’être ouvert pour emballer.
- Un `Emballage` contenant quelque chose ne peut emballer.
- L’elfe peut ouvrir un `Emballage` pour accéder à ce qui a été emballé (méthode `ouvreMoi` ).
- Une fois l’`Emballage` ouvert et que l’elfe a pris l’objet, l’`Emballage` est de nouveau vide.
- L’elfe peut fermer un `Carton` ouvert ( `fermeMoi` )
- Quand l’elfe prends un emballage il dit “siffle en travaillant”
- Quand l’elfe emballe il dit “tuuuut tuuut tuut”

Il vous demande d’écrire les classes `Emballage` , `Carton` , `PapierCadeau`.

Il veut que vous implémentiez la fonction:

```
1 Objet*MesTestUnitaires(Objet **);
```

Il vous donne un tableau de 3 éléments (élément nul pour fin), contenant respectivement 1 nounours, 1 carton, et 1 papier cadeau. Vous devez tout emballer et retourner un cadeau.

Les cas d'erreurs doivent générer des messages explicites sur la sortie d'erreur. Libre à vous d'effectuer plus de programmes de test de votre côté.

## Chapitre IV

# Tapis qui roule n'amasse pas mousse

Il faut maintenant être capable de gérer le poste de travail :

- L'elfe a une `Table` devant lui et un `TapisRoulant` à côté de lui.
- On ne peut rien poser sur le tapis roulant s'il y a déjà quelque chose.
- L'elfe peut Poser et Prendre sur la table les `Objet`.
- Idem sur le tapis roulant.
- Quand il n'y a plus de place sur la table, elle craque. Elle peut contenir un maximum de 10 objets.
- L'elfe reçoit des `Emballages` en appuyant sur le bouton "IN" du `TapisRoulant` avec une `Main`, ou envoie ce qu'il y a actuellement sur le tapis au père Noël en appuyant sur le bouton "OUT".
- Un `Emballage` envoyé au père Noël est considéré comme disparu, le `TapisRoulant` est de nouveau libre.
- L'elfe peut `Regarder` sur la `Table` pour savoir ce qu'il s'y trouve. Il obtient un tableau des intitulés des différents `Objets`. Le dernier élément du tableau est à `null`.
- L'elfe peut `Regarder` sur le tapis roulant pour savoir ce qu'il s'y trouve.
- C'est une erreur d'appuyer sur un bouton du tapis roulant si l'entrée/sortie du tapis n'a pas été initialisée.

Il vous demande d'écrire les interfaces `ITable` et `ITapisRoulant` et les classes `TablePePeNoel` et `TapisRoulantPePeNoel` implémentant ces interfaces. La table et le tapis roulant du père Noël contiennent ce qu'il faut pour faire 2 cadeaux. La disposition/répartition des emballages/jouets entre le tapis et la table est à votre convenance.

Il veut que vous lui fournissiez deux fonctions :

```
1 ITable *creerTable();  
2 ITapisRoulant *creerTapisRoulant();
```

permettant de réifier (instancier) ces 2 objets. Les cas d'erreurs doivent générer des messages explicites sur la sortie d'erreur.



## Chapitre V

# Travail d'elfe (ou pourquoi je ne ferais de liste de Noël)



Il faut maintenant gérer l'elfe.

Il vous demande d'écrire l'interface `IElf` implémentant toutes les actions d'un elfe tel que décrit dans l'ensemble du sujet.

Ensuite implémenter un `ElfDePePeNoel` implémentant cette interface.

Quand on demande à l'elfe d'emballer un cadeau il se débrouille, entre la table et le tapis roulant pour emballer un cadeau et l'envoyer par le tapis roulant par un `OUT`.

## Chapitre VI

# Tiens v'la la boîte vieux schnock



Shadokons un peu les choses. Implémentez une classe `TableAleat` et `TapisRoulantAleat` qui présente une disposition aléatoire d'objets. Votre elfe doit être capable de fabriquer le maximum de cadeaux possible et de l'envoyer au père Noël.

Quand il ne peut plus rien faire (pas de quoi faire un cadeaux), il hurle:

```
1 pepe ya un schmolle dans le bignou
```

De plus `TapisRoulantAleat` a un comportement bizarre, et fait beaucoup de bruit. Le goblin mécano dit qu'il cause. "Li coz'l'XML" comme il dit. Enfin bon avec son accent gobelin on comprend rien.

Toujours est-il qu'il faut qu'li coz'l'XML votre tapis. C'est à dire que le tapis décrit le paquet que vous envoyer en commençant de l'emballage vers l'intérieur en XML. 'Sérialisation' qu'il appelle ca le goblin. Pas de panique, c'est un goblin. Quand on appuie sur le bouton couleur vert Zwify, le `TapisRoulantAleat` crache une fiche XML.

## Chapitre VII

# Le Père Noël est-il quantique ?\*

\*ben oui quoi, livrer tous les jouets en même temps tout autour de la terre! C'est quoi cette ubiquité à 2 balles?

Il faut maintenant implémenter un père Noël qui est un programme à part. Le père Noël est fainéant et attend des paquets pour les mettre dans le traineau. Un paquet est le résultat d'une sérialisation de vos objets en XML. Ces fichiers sont donnée au père Noël en paramètre sur la ligne de commande.

```
1 > ./perenoel cadeau1.xml cadeau2.xml cadeau3.xml
```

Vous devrez PROUVER à l'inspecteur des travaux finis que votre père Noël charge bien chaque cadeau et recréer bien ceci en mémoire en procédant à ce que ce funeste gobelin appelle 'désérialisation'. Pour cela vous devrez prendre chaque cadeau, l'ouvrir, prendre le jouet à l'intérieur et l'y remettre. Et oui, le contrôle qualité c'est important pour le père Noël. Lui c'est pas un idiot : pour savoir si le chat est mort, il ouvre la boîte! Et prendre l'objet à l'intérieur ca va faire gueuler le jouet, ce qui permettra de savoir ce qu'il y a à l'intérieur.



Ce qui suis est plus destiné à du tech3s. Toutefois les vrais guerriers peuvent s'y frotter.

## Chapitre VIII

# The Warp Machine ou un degré de plus avant la démente.



Vision conceptuelle de la WarpMachine

Les paquets proviennent d'une machine appelée "The Warp Machine". Tout paquet envoyé par le nouveau tapis avec la commande OUT est reçu par la warp machine, par une de ses parties qui ressemble à une grosse chaussette magique. Un peu comme celles qu'on pend au dessus de la cheminée mais en vachement plus moche. Ce sont les nains Porte et PaNawak qui nous l'ont donné il y a très longtemps contre l'intégrale de Tommy verdâtre "chanson pour corps caverneux". Toujours est-il que cette sacrée machine attend un nombre mystique pour être reliée au tapis magique des elfes. Nombre qu'il faudra bien évidemment donner au père Noël quand on le lancera en mode warp avec un paramètre '-w' comme «Warp Me Up Scotty».

Le chef des deux nains a expliqué au père Noël comment écrire ce nombre. Voici ce qui a été dit...\*



Pour relier la chaussette magique au warp,  
il faut que cette chaussette soit 'figuré en 'Hue de pet multi-casque!?'  
Le nombre mystique marche par quadruplet.  
Le premier commence toujours entre 224 et 239.  
Demande à mon cousin gogle pour plus d'infos.

\*enfin après traduction approximative de son dialecte «le shmolesque du nord».

Voilà. Modifiez votre programme père Noël pour qu'il prenne ce nombre mystique en paramètre et qu'il configure convenablement la warp machine en ayant relié la chaussette magique. De plus, le gobelin mécano a fait une nouvelle version du `TapisRoulantAleat` : «le TapisMagique». Comme le `TapisRoulantAleat` Li coz'l'XML et ce bruit est transmis dans le warp. Cet XML est capté par la chaussette et est joué par une partie de la warp machine qui ressemble à un gramophone au père Noël. Le père Noël sait donc ce qui est produit dans le tapis. Le père Noël grâce à cette musique fait surgir le paquet du Warp. Et il le prouve à l'inspecteur en saisissant vigoureusement le jouet qui se trouve à l'intérieur.

Grâce à cela plusieurs chaines de production elfe peuvent envoyer leurs paquets au père Noël.

Si plusieurs père Noël utilisent le même nombre mystique ils reçoivent tous les cadeaux destinés à ce nombre mystique ce qui est très pratique pour dupliquer les cadeaux. Plusieurs père Noël? Mais on a dit qu'il était quantique...suivez enfin!



Bonus: Essayez d'imposer la langue de l'XML aux autres groupes. Pour faire travailler des elfes de différents groupes avec différents père Noël.