

UNIVERSITÉ DE SHERBROOKE
DÉPARTEMENT D'INFORMATIQUE

IFT 159

Devoir #1 — Automne 2009

Analyse et conception

Devoir à remettre au plus tard le 1^{er} octobre 2009 à 23h59. La remise devra être faite grâce à la commande *turnin* (voir à la fin de l'énoncé pour plus de détails).

Le but de ce travail consiste à implanter la solution d'un problème dont vous recevez la spécification, l'analyse et la conception. L'objectif est de vous familiariser, au niveau programmation, avec la notion de séquence et de fonctions. **Vous devez obligatoirement implanter la conception fournie en annexe en utilisant seulement les notions de programmation vues dans le cours. L'utilisation de structures sélectives ou itératives n'est pas permise..**

Vous devez me remettre votre programme, trois fichiers tests ainsi que la documentation. Votre programme devra respecter les normes de programmation du département. Les fichiers tests devront contenir des tests significatifs. La documentation se fera sous forme de commentaires dans le programme en utilisant les normes de *doxygen*.

Sujet : Document latex pour le classement des athlètes.

Aux olympiades, le juge en chef des différentes compétitions (plongeon, gymnastique, ...) désire produire un document de format PDF contenant un tableau pour le classement des 4 athlètes qui participent à chacune des compétitions. Pour chacune des compétitions, on veut produire un document PDF. Le tableau de classement contiendra le nom de l'athlète, son numéro, les notes des 3 juges et la moyenne des notes. On veut aussi avoir la moyenne des notes accordées par chacun des juges ainsi que la moyenne des "notes moyennes" obtenues par les athlètes.

Pour produire son fichier PDF contenant les résultats, le juge en chef ne dispose que du logiciel de traitement de texte « latex ». Celui-ci ne possède pas d'interface usager convivial. On vous demande donc de développer un programme qui reçoit les résultats de la compétition en entrée et qui génère automatiquement le code latex requis.

Les résultats des athlètes se trouvent dans un fichier. Le format de ce fichier est donné à la figure 1. Vous trouverez à la figure 2 un exemple d'affichage des résultats dans un document PDF.

```

« Numéro de l'athlète 1 »
« Résultat du premier juge »
« Résultat du second juge »
« Résultat du troisième juge »


« Numéro de l'athlète 2 »
« Résultat du premier juge »
« Résultat du second juge »
« Résultat du troisième juge »

« Numéro de l'athlète 3 »
« Résultat du premier juge »
« Résultat du second juge »
« Résultat du troisième juge »

« Numéro de l'athlète 4 »
« Résultat du premier juge »
« Résultat du second juge »
« Résultat du troisième juge »

```

FIG. 1 – Format du fichier contenant les résultats des athlètes.



UNIVERSITÉ DE

SHERBROOKE

Olympiades 2004

Résultat du plongeon

Numéro de l'athlète	Note juge 1	note juge 2	note juge 3	moyennes
11	5	8	9	7.33
22	9	9	9	9
53	6	6	7	6.33
Moyenne par juge	6.67	7.66	8.33	7.55

FIG. 2 – Exemple de document PDF affichant les résultats.

Introduction à latex

Écrire un document latex consiste tout d'abord à spécifier le format de la page puis à insérer le contenu du document. Le contenu du document peut être composé de texte, de tableaux, d'images et autres éléments. Tous ces éléments sont formatés grâce à des balises. Ces balises sont le plus souvent des commandes pour indiquer la taille des caractères, la couleur du texte, etc...

Ces balises se présentent souvent sous la forme : `\begin{balise}` et `\end{balise}`. Tout ce qui se trouve entre le « begin » et le « end » de la balise est affecté par celle-ci. Une balise peut aussi s'écrire sous la forme `{\balise texte}`. Le texte entre `{` et `}` est affecté par la balise.

La structure d'un document latex

Pour écrire un document latex, mieux vaut d'abord en connaître la structure. Celle-ci est très simple. Un document latex est composé :

- d'une entête qui sert à décrire :
 - le type de document (lettre, A4, ...),
 - sa forme (simple ou double colonnes, ...),
 - le format de la page (la taille des marges, l'interligne, ...)
 - la taille des caractères,
 - les différents « package » à utiliser.
- du corps du document, inclus entre `\begin{document}` et `\end{document}` qui contient le texte lui-même.

Un exemple de document est donné à la figure 3. Le document produit par ce fichier est donné à la figure 4.

Les balises de latex

Les principales balises servent à :

- centrer un document (`\begin{center}` et `\end{center}`);
- insérer un tableau (`\begin{tabular}{clr|c}` et `\end{tabular}`);

Les paramètres associés à un tableau sont le nombre de colonnes avec la justification dans les colonnes. Dans la déclaration ci-haut, il y a quatre colonnes. Le texte est centré (« c ») dans la première et dernière colonne, justifié à gauche (« l ») dans la seconde colonne et justifié à droite (« r ») dans la dernière. La barre verticale (« | ») entre deux colonnes sert à les séparer d'un trait vertical. Pour séparer les lignes d'un trait, il faut utiliser « `\hline` ». Dans le texte, le « & » sert à séparer les colonnes et le « `\\` » les lignes.

- insérer une figure (`\begin{figure}[H]` et `\end{figure}`);

Les paramètres associés à une figure concernent sa position dans le document. Le paramètre spécifié ci-haut (H pour here) indique que l'on doit mettre la figure là où elle se trouve dans le texte original.

- modifier la taille des caractères (`\tiny`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, ...)
- modifier l’aspect des caractères (`\bf`, `\it`, `\tt`, ...)

Il existe beaucoup d’autres balises. Pour des descriptions plus complètes de latex et de ses balises, consultez les sites Web suivants :

- <http://www.tug.org.in/tutorials.html>
- <http://www.maths.tcd.ie/~dwilkins/LaTeXPrimer/>
- <http://it.metr.ou.edu/latex/>
- <http://www.cs.cornell.edu/Info/Misc/LaTeX-Tutorial/LaTeX-Home.html>
- <http://heather.cs.ucdavis.edu/~matloff/latex.html>
- <http://osiris.sunderland.ac.uk/~cs0pdu/pub/mat118/latexut.htm>
- <http://www.sci.usq.edu.au/staff/robertsa/LaTeX/latexintro.html>
- <http://www.docsdunet.com/cours/faq-tex-french.html>

Analyse

Entrées :

- Nom de la compétition (chaîne de caractères)
- Type de compétition (chaîne de caractères)
- Nom du fichier contenant les résultats (chaîne de caractères)
- Numéro des athlètes (entiers)
- Résultats des athlètes (réels)

Sorties :

- Document latex contenant le tableau des résultats et moyennes (chaînes de caractères).
- Nombre de lignes produites dans le fichier de sortie.

Constantes :

- Nombre de juges = 3
- Nombre d'athlètes = 4

Formules :

- Moyenne des notes = $\frac{\text{Note juge 1} + \text{Note juge 2} + \text{Note juge 3}}{\text{nombre de juges}}$
- Moyenne par juge = $\frac{\sum_{i=1}^4 \text{Note athlète } i}{\text{Nombre d'athlètes}}$
- Moyenne des moyenne = $\frac{\sum_{i=1}^4 \text{Moyenne des notes}_i}{\text{nombre de d'athlètes}}$

Notes importantes :

1. Dans la conception, nous utilisons le terme générer dans deux situations différentes :
 - (a) *Générer xxxx (module yyy)* signifie que l'on utilise une fonction pour effectuer l'action.
 - (b) *Générer zzzz ("begin{fff}")* signifie que l'on écrit directement la chaîne de caractères dans le fichier.
2. Le nom du fichier destination doit être complet. Un fichier latex termine avec l'extension *tex*. Vous devez donc entrer *nomDuFichier.tex*.

Conception

Conception globale (diagramme structurel)

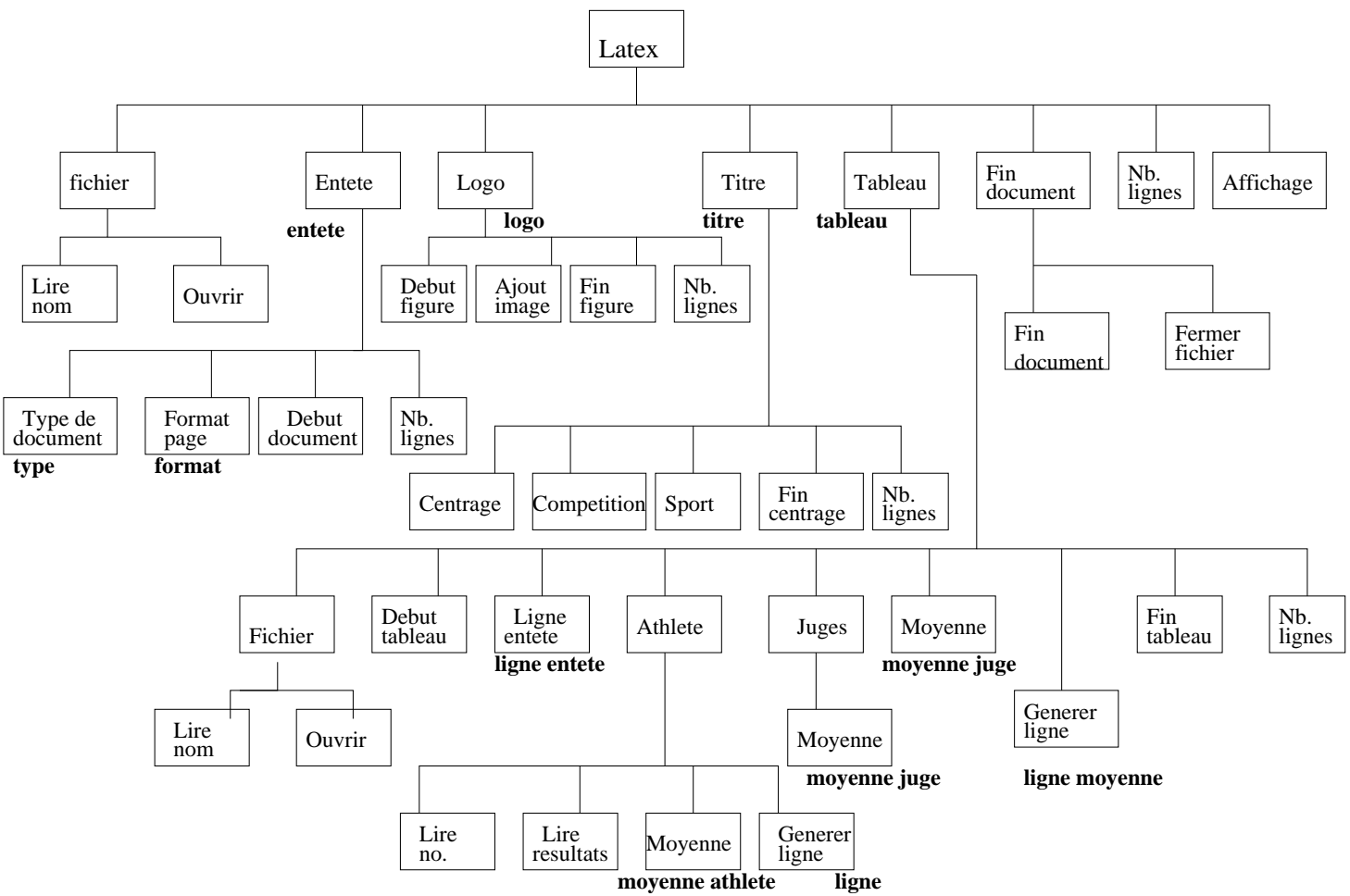


FIG. 5 – diagramme structurel

MODULE *Latex*

Analyse

Entrée : (lecture) Nom du fichier de sortie (chaîne de caractères))

Sortie :

- fichier de sortie (fichier)
- nombre de lignes dans le fichier (entier)

Conception (algorithme)

1. Fichier de sortie
 - (a) lire le nom
 - (b) ouverture en écriture
 2. Générer l'entête (module Entête)
 3. Cumuler le nombre de lignes
 4. Générer le logo (module Logo)
 5. Cumuler le nombre de lignes
 6. Générer le titre (module Titre)
 7. Cumuler le nombre de lignes
 8. Générer le tableau (module Tableau)
 9. Cumuler le nombre de lignes
 10. Générer fin document ("`\end{document}\n`")
 11. Incrémenter le nombre de lignes
 12. Fermer du fichier de sortie
 13. Afficher le nombre de lignes
-

MODULE *Entête*

Analyse

Entrée : (paramètre) Fichier de sortie (fichier)

Sortie :

- (paramètre) fichier de sortie (entier)
- (retour) nombre de lignes (entier)

Conception

1. Générer type de document (module Type)
2. Cumuler le nombre de lignes
3. Générer format pages (module Format page)
4. Cumuler le nombre de lignes
5. Générer debut document ("`\begin{document}\n`")
6. Incrémenter le nombre de lignes

MODULE *Logo*

Analyse

Entrée : (paramètre) Fichier de sortie (fichier)

Sortie :

- (paramètre) fichier de sortie (entier)
- (retour) nombre de lignes (entier)

Conception

1. Générer début figure ("`\begin{figure}[H]\n`")
 2. Incrémenter le nombre de lignes
 3. Générer inclusion de la figure ("`\epsfig{file=logoudes.eps}\n`")
 4. Incrémenter le nombre de lignes
 5. Générer fin de la figure ("`\end{figure}\n`")
 6. Incrémenter le nombre de lignes
-

MODULE *Titre*

Analyse

Entrée :

- (paramètre) fichier de sortie (fichier)
- (lecture) nom de la compétition (chaînes de caractères)
- (lecture) type de sport (chaînes de caractères)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception

1. Générer centrage ("`\begin{center}\n\vspace{2cm}\n`")
2. Augmenter de 2 le nombre de lignes
3. Compétition :
 - (a) lire le nom
 - (b) générer format dans fichier ("`{\LARGE\bf` ")
 - (c) écrire le nom compétition dans fichier
 - (d) générer fin format dans fichier ("`}\n\n`")
 - (e) augmenter de 2 le nombre de lignes
4. Sport :
 - (a) lire le type
 - (b) générer format dans fichier ("`\vspace{1cm}\n{\large\it` ")
 - (c) écrire le type dans fichier
 - (d) générer fin format dans fichier ("`}\n\n`")
 - (e) augmenter de 3 le nombre de lignes
5. Générer fin centrage ("`\end{center}\n\vspace{1cm}\n`")
6. Augmenter de 2 le nombre de lignes

MODULE *Tableau*

Analyse

Entrée :

- (paramètre) fichier de sortie (fichier)
- (lecture) nom du fichier de résultats (chaîne de caractères)
- (lecture) numéro des athlètes (entier)
- (lecture) résultats des athlètes (réels)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception

1. Fichier de résultats :
 - (a) lire le nom
 - (b) ouvrir en lecture
2. Générer début du tableau (`"\begin{tabular}[[c|c|c|c|c|]\n\hline\n"`)
3. Augmenter de 2 le nombre de lignes
4. Générer ligne entête (module Ligne entête)
5. Cumuler le nombre de lignes
6. Athlète 1 :
 - (a) lire le numéro et les trois résultats
 - (b) calculer moyenne (module Moyenne athlète)
 - (c) générer ligne (module Ligne)
 - (d) cumuler le nombre de lignes
7. Athlète 2 :
 - (a) lire le numéro et les trois résultats
 - (b) calculer moyenne (module Moyenne athlète)
 - (c) générer ligne (module Ligne)
 - (d) cumuler le nombre de lignes
8. Athlète 3 :
 - (a) lire le numéro et les trois résultats
 - (b) calculer moyenne (module Moyenne athlète)
 - (c) générer ligne (module Ligne)
 - (d) cumuler le nombre de lignes
9. Athlète 4 :
 - (a) lire le numéro et les trois résultats
 - (b) calculer moyenne (module Moyenne athlète)
 - (c) générer ligne (module Ligne)
 - (d) cumuler le nombre de lignes

10. juges
 - (a) calculer moyenne juge 1 (module Moyenne juge)
 - (b) calculer moyenne juge 2 (module Moyenne juge)
 - (c) calculer moyenne juge 3 (module Moyenne juge)
 11. Calculer moyenne des moyennes (module Moyenne juge)
 12. Générer ligne moyenne (module Ligne moyenne)
 13. Cumuler le nombre de lignes
 14. Générer fin du tableau (`"\end{tabular}\n"`)
 15. Incréments le nombre de lignes
-

MODULE *Type*

Analyse

Entrée : (paramètre) fichier de sortie (fichier)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception

1. Générer ligne document (`"\documentclass[12pt]{article}\n"`)
 2. Incréments le nombre de lignes
 3. Générer langue du document (`"\usepackage[frenchb]{babel}\n"`)
 4. Incréments le nombre de lignes
 5. Générer le support pour caractères accentués (`"\usepackage[latin1]{inputenc}\n"`)
 6. Incréments le nombre de lignes
 7. Générer le support pour les images de type « eps » (`"\usepackage{epsfig}\n"`)
 8. Incréments le nombre de lignes
-

MODULE *Format page*

Analyse

Entrée : (paramètre) fichier de sortie (fichier)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception

1. Générer largeur document (`"\setlength{\textwidth}{6.5in}\n"`)
2. Incréments le nombre de lignes
3. Générer hauteur document (`"\setlength{\textheight}{9.5in}\n"`)
4. Incréments le nombre de lignes

MODULE *Ligne entête*

Analyse

Entrées : (paramètre) fichier de sortie (fichier)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception (algorithme)

1. Générer cellule (écrire dans le fichier) "Numéro de l'athlète" et fin cellule ("&")
 2. Générer cellule "Note juge 1" et fin cellule
 3. Générer cellule "Note juge 2" et fin cellule
 4. Générer cellule "Note juge 3" et fin cellule
 5. Générer cellule "Moyenne" et fin ligne ("\\n\\hline\\n\\hline\\n")
 6. Augmenter de 3 le nombre de lignes
-

MODULE *Moyenne athlète*

Analyse

Entrées : (paramètre) les 3 notes de l'athlète (réels)

Sortie : (retour) la moyenne des notes (réel)

Constante : nombre de juges = 3

Formule : Moyenne des notes = $\frac{\text{Note juge 1} + \text{Note juge 2} + \text{Note juge 3}}{\text{nombre de juges}}$

Conception (algorithme)

1. Calculer la moyenne (formule)
-

MODULE *Moyenne juge*

Analyse

Entrées : (paramètre) 4 valeurs (réels)

Sortie : (retour) la moyenne des valeurs (réel)

Constante : nombre d'athlètes = 4

Formule : Moyenne des moyenne = $\frac{\sum_{i=1}^4 \text{Valeur}_i}{\text{nombre d'athlètes}}$

Conception (algorithme)

1. Calculer la moyenne (formule)

MODULE *Ligne*

Analyse

Entrées :

- (paramètre) fichier de sortie (fichier)
- (paramètre) numéro de l'athlète (entier)
- (paramètre) résultat 1 (réel)
- (paramètre) résultat 2 (réel)
- (paramètre) résultat 3 (réel)
- (paramètre) moyenne des résultats (réel)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception (algorithme)

1. Générer cellule avec le numéro et fin cellule ("&")
2. Générer cellule avec résultat 1 et fin cellule
3. Générer cellule avec résultat 2 et fin cellule
4. Générer cellule avec résultat 3 et fin cellule
5. Générer cellule avec moyenne des résultats et fin ligne ("\\n\\hline\\n")
6. Augmenter de 2 le nombre de lignes

MODULE *Ligne moyenne*

Analyse

Entrées :

- (paramètre) fichier de sortie (fichier)
- (paramètre) moyenne 1 (réel)
- (paramètre) moyenne 2 (réel)
- (paramètre) moyenne 3 (réel)
- (paramètre) moyenne des moyennes (réel)

Sortie :

- (paramètre) fichier de sortie (fichier)
- (retour) nombre de lignes (entier)

Conception (algorithme)

1. Générer cellule "Moyenne par juge" et fin cellule ("&")
2. Générer cellule avec moyenne 1 et fin cellule
3. Générer cellule avec moyenne 2 et fin cellule
4. Générer cellule avec moyenne 3 et fin cellule
5. Générer cellule avec moyenne des moyennes et fin ligne ("\\n\\hline\\n")
6. Augmenter de 2 le nombre de lignes

Informations complémentaires

1. Vous trouverez l'image à utiliser dans le document dans le fichier
`/home/public/cours/ift159/logoudes.eps`.
Vous devez le copier dans votre répertoire de travail afin de pouvoir l'utiliser.
2. Vous pouvez compiler votre fichier latex afin de produire un fichier PDF et valider vos résultats. Les commandes sont :

```
latex xxxx.tex
dvi2pdf xxx
```

Cette dernière commande produit le fichier `xxx.pdf`. Vous pouvez visionner le fichier PDF avec « `acroread` ».

3. Les noms de fichier ne doivent pas être de type "string". Ils doivent être déclarés comme des tableaux de caractères. La déclaration se fait de la façon suivante :

```
char nomFichier[30] ;
```

Un tel tableau peut contenir une chaîne d'au plus 30 caractères. Vous pouvez lire une chaîne de caractères directement dans le tableau (`cin >> nomFichier;`).

4. Lorsque nous parlons de fichiers en paramètre, il s'agit bien d'un `ifstream` ou `ofstream` que vous devez passer à l'implantation. Pour passer un `ifstream` ou `ofstream` en paramètre, vous devez faire :

```
déclaration : int cellule(ofstream& , ifstream&) ;
définition : int cellule(ofstream& fichierSortie, ifstream& fichierResul)
appel de la fonction :   ofstream fichierTEX ;
                        ifstream fichierRes ;
                        int resultat ;

                        resultat = cellule(fichierTEX, fichierRes) ;
```

5. **Note importante** : afin d'afficher un `\` dans une chaîne de caractères en C++, il doit être doublé.

Soumission

Vous devez me soumettre avant la date limite, à l'aide de la commande

```
turnin -cift159 -ptp2 tp2.cpp test1 test2 test3 calculeffort.txt
```

un seul fichier portant OBLIGATOIREMENT le nom `tp2.cc` qui contient le programme demandé et 3 fichiers de tests. Les trois tests soumis doivent être significatifs, i.e. qu'ils permettent de tester le plus de cas possibles pour votre application.

Étant donné que la correction est partiellement automatisée, une mauvaise soumission entraîne une mauvaise note (possiblement 0). Il n'y a pas de recorection.

Pour la bonne utilisation du train de travail ultérieur à votre remise utiliser "endl" plutôt que "`\n`" pour changer de ligne.