



UNIVERSITÉ DE MONCTON
CAMPUS DE SHIPPAGAN

Projet (25% de la note finale)

INFO1101 Principes de programmation I

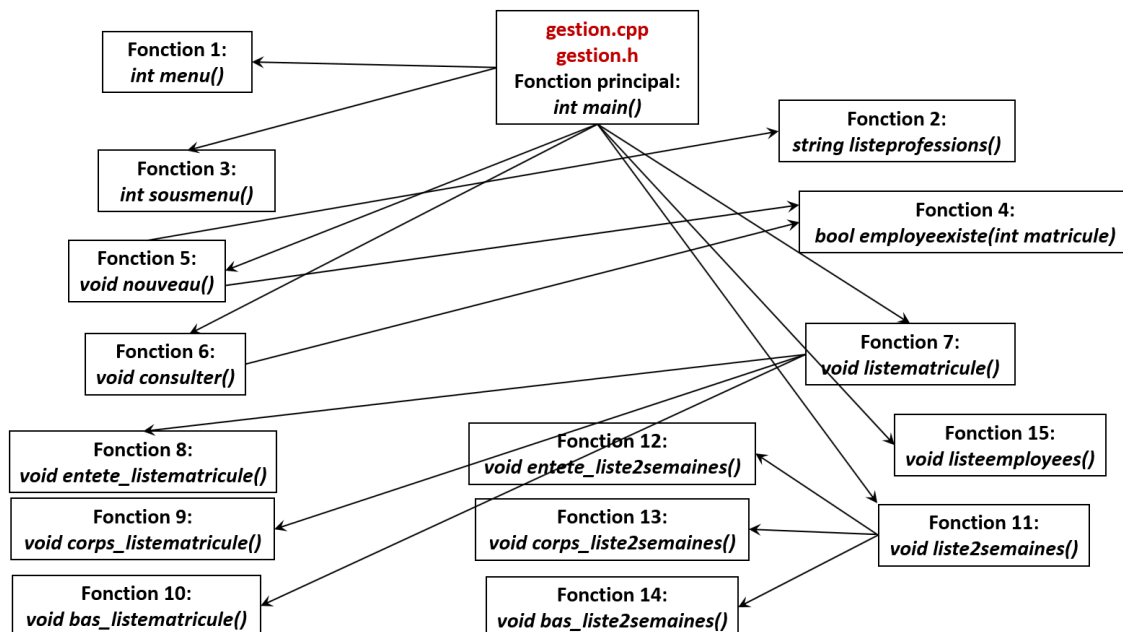
Date d'échéance : le samedi 9 décembre 2023

PÉNALITÉ DE RETARD

Le travail ne sera pas accepté après 23h59.

Déposer vos réponses sur clic UMCS dans la boîte "Projet" sous "Évaluation-Rendement".

Le travail demandé consiste à compléter le projet de gestion des employé-e-s dont le schéma d'analyse selon l'approche descendante (*Top-Down*) est donné par la figure ci-dessous.



Consignes et tâches :

- 1- Le code du projet est constitué de **Projet.sln**, **Projet.vcxproj**, **Projet.vcxproj.filters**, **Projet.vcxproj.user**, **gestion.cpp** et **gestion.h**.
- 2- Ouvrir la solution **Projet.sln** et compléter le code de **gestion.cpp**.
- 3- Le code des fonctions `string listeprofessions()`, `bool employeexiste(int matricule)`, `void listematricule()`, et `void liste2semaines()` est fourni. Essayez de le comprendre pour savoir l'utiliser. Vous n'avez pas besoin de le changer, mais vous pouvez le faire si vous le voulez.
- 4- Complétez le code de toutes les autres fonctions incluant `int main()`.
- 5- Les variables globales suivantes sont déclarées dans **gestion.h** :
`string Vemplois[5]`, `int mat`, `string nom`, `int annee`, `float salaireh`, `string profession`, `float montant` et `float total` et `int numero`.
 - Vous pouvez les utiliser dans n'importe quelle fonction sans les redéclarer comme je l'ai déjà fait dans `bool employeexiste(int matricule)`.
 - Vous avez aussi le droit de déclarer d'autres variables au besoin.
- 6- La fonction `int menu()` :
 - afficher un menu.
 - lire le choix.
 - retourner le choix.
- 7- La fonction `int sousmenu()` :
 - afficher le menu du choix 3) **Imprimer les etats**.
 - lire le choix.
 - retourner le choix.
- 8- La fonction `bool employeexiste(int matricule)` : **est déjà programmée**
 - Elle lit les informations du fichier **"personnel.dat"** : cinq lignes par employé-e.
 - Elle vérifie si la 1ere ligne est égale au paramètre `matricule`
 - Si oui elle retourne `true` (c.à.d. trouvé : existe).
 - Sinon elle retourne `false` (c.à.d. **NON** trouvé : n'existe pas).
 - Cette fonction est appelée par `void nouveau()` et `void consulter()` seulement.
- 9- La fonction `void nouveau()` :
 - lire une matricule.
 - Appeler `bool employeexiste(int matricule)` pour vérifier si cette matricule est déjà assignée à un-e employé-e.
 - Si oui afficher le message **"ERREUR : Employe-e existant-e ..."**.
 - Sinon :
 - lire les autres informations : nom, annee, salaireh
 - Appeler `listeprofessions()` pour choisir une profession.
 - Ajouter les informations saisies au fichier **"personnel.dat"**

10- La fonction `void consulter()` :

- lire une matricule.
- Appeler `bool employeexiste(int matricule)` pour vérifier si cette matricule est déjà assignée à un-e employé-e.
- Si oui afficher les autres informations de l'employé-e : nom, annee, salaireh, profession du fichier `"personnel.dat"`
- Sinon afficher le message `"ERREUR : Employe-e inexistant-e ..."`.

11- La fonction `void listematricule()` : est déjà programmée

- Imprime l'état `"listematricule.dat"` en appelant :
 - `entete_listematricule();`
 - `corps_listematricule();`
 - `bas_listematricule();`

12- La fonction `void entete_listematricule()` :

- Imprime l'entête de l'état `"listematricule.dat"`

```
1 | LISTE DES EMPLOYE-ES
2 +-----+-----+-----+-----+
3 |Numero|Matricule| Nom          |Annee |
4 |-----|-----|-----|-----|
```

13- La fonction `void corps_listematricule()` :

- Imprime le corps de l'état `"listematricule.dat"`

```
5 | 1 | 351 | Dupont          | 2002 |
6 | 2 | 29  | Jean Talon         | 1999 |
7 | 3 | 450 | Aghate             | 2014 |
8 | 4 | 112 | Julie              | 2005 |
9 | 5 | 901 | Steve              | 2002 |
10| 6 | 90  | Mohammed Sidi      | 2011 |
11| 7 | 999 | Peter Dan          | 2018 |
12| 8 | 201 | Yves               | 2000 |
```

14- La fonction `void bas_listematricule()` :

- Imprime le bas de l'état `"listematricule.dat"`

```
13 +-----+-----+-----+-----+
```

15- La fonction `void liste2semaines()` : est déjà programmée

- Imprime l'état `"paie.dat"` en appelant :
 - `entete_liste2semaines();`
 - `corps_liste2semaines();`
 - `bas_liste2semaines();`

16- La fonction `void entete_liste2semaines()` :

- Imprime l'entête de l'état `"paie.dat"`

```
1 | PAIE DES EMPLOYE-ES : DEUX SEMAINES
2 +-----+-----+-----+-----+-----+
3 |Numero|Matricule| Nom          |Salaire horaire|Paie de 2 semaines|
4 |-----|-----|-----|-----|-----|
```

17- La fonction `void corps_liste2semaines()` :

- Imprime le corps de l'état `"paie.dat"`

5		1		351		Dupont		21.50		1526.50	
6		2		29		Jean Talon		45.00		3195.00	
7		3		450		Aghate		15.75		1118.25	
8		4		112		Julie		30.25		2147.75	
9		5		901		Steve		11.50		816.50	
10		6		90		Mohammed Sidi		12.78		907.38	
11		7		999		Peter Dan		12.75		905.25	
12		8		201		Yves		35.00		2485.00	

18- La fonction `void bas_liste2semaines()` :

- Imprime le bas de l'état `"paie.dat"`

13		-----		-----		-----		-----		-----	
14								Total =		13101.63	
15		+		-----		-----		-----		-----	

19- La fonction `void listeemployees()` :

- Imprime l'état `"employees.dat"`

- N'utilisez pas des fonctions `entete()`, `corps()` et `bas()`, mais écrivez les directement dans cette fonction.

1		=====		=====		=====		=====		=====	
2		EMPLOYE-E		1		Dupont					
3		=====		=====		=====		=====		=====	
4		Matricule		:		351					
5		-----		-----		-----		-----		-----	
6		Profession		:		Enseignante/Enseignant					
7		-----		-----		-----		-----		-----	
8		Annee de recrutement		:		2002					
9		-----		-----		-----		-----		-----	
10		Salaire horaire		:		21.50					
11		-----		-----		-----		-----		-----	
12		Salaire d'une semaine		:		763.25					
13		=====		=====		=====		=====		=====	
14		=====		=====		=====		=====		=====	
15		EMPLOYE-E 2		:		Jean Talon					
16		=====		=====		=====		=====		=====	
17		Matricule		:		29					
18		-----		-----		-----		-----		-----	
19		Profession		:		Rectrice/Recteur					
20		-----		-----		-----		-----		-----	
21		Annee de recrutement		:		1999					
22		-----		-----		-----		-----		-----	
23		Salaire horaire		:		45.00					
24		-----		-----		-----		-----		-----	
25		Salaire d'une semaine		:		1597.50					
26		=====		=====		=====		=====		=====	
27		=====		=====		=====		=====		=====	
28		EMPLOYE-E 3		:		Aghate					

NOTE :

- Voir les fichiers exemples fournis (**personnel.dat**, **listematricule.dat**, **employees.dat**, **paie.dat**, **employees.dat**) pour avoir une idée.
- Si vous voulez voir l'exécution du programme C++ demandé :
 - o Créer un dossier **ProjetResultat** ou autre dans votre ordinateur.
 - o Copier le **projet.unk** dans ce dossier.
 - o Changer le nom du fichier **projet.unk** à **projet.exe**.
 - o Exécuter le fichier **projet.exe** et tester le :
 - Ajouter quelques employé-e-s (choix **1**) du menu principal).
 - Consulter un-e employé-e par numéro de matricule (choix **2**)).
 - Tester le choix 3) du menu principal.
 - Après ces tests, le programme **projet.exe** créera les fichiers **personnel.dat**, **listematricule.dat**, **employees.dat**, et **paie.dat** dans le dossier **ProjetResultat**. Vous pouvez ouvrir chaque fichier avec un éditeur texte pour voir son contenu. Vous pouvez aussi les voir avec Visual Studio comme fichier texte.

Un exemple d'exécution :

```
=====
Menu :      Gestion du personnel
=====
1) Recruter un nouveau/une nouvelle employe-e
2) Consulter les informations d'un-e employe-e
   -----
3) Imprimer les etats
   -----
4) Quitter l'application
-----
Choix -> 1

-> Nouveau/nouvelle employe-e   :
-----
Matricule      = 999

ERREUR : Employe-e existant-e ...

=====
Menu :      Gestion du personnel
```

- ```
=====
1) Recruter un nouveau/une nouvelle employe-e
2) Consulter les informations d'un-e employe-e
```

```

3) Imprimer les etats
```

```

4) Quitter l'application
```

```

Choix -> 1
```

```
-> Nouveau/nouvelle employe-e :
```

```

Matricule = 200
Nom = Rachel Dyan
Annee de recrutement = 2001
Salaire horaire (entre 11.50 et 75.00) = 76
Salaire horaire (entre 11.50 et 75.00) = 15.75
```

```

 Liste des professions
```

- ```
-----
1) Rectrice/Recteur
2) Doyenne/Doyen
3) Enseignante/Enseignant
4) Agente/Agent de bureau
5) Technicienne/Technicien
```

```
-----
Profession -> 3
```

```
=====
Menu :      Gestion du personnel
```

- ```
=====
1) Recruter un nouveau/une nouvelle employe-e
2) Consulter les informations d'un-e employe-e
```

```

3) Imprimer les etats
```

```

4) Quitter l'application
```

```

Choix -> 2
```

-> Consulter les informations d'un-e employe-e :

-----  
Matricule = 200

=====

Employe-e

=====

Matricule : 200

-----  
Nom : Rachel Dyan

-----  
Annee de recrutement : 2001

-----  
Salaire horaire : 15.75

-----  
Profession : Enseignante/Enseignant

=====

=====

Menu : Gestion du personnel

=====

1) Recruter un nouveau/une nouvelle employe-e

2) Consulter les informations d'un-e employe-e

-----  
3) Imprimer les etats

-----  
4) Quitter l'application

-----  
Choix -> 2

-> Consulter les informations d'un-e employe-e :

-----  
Matricule = 989

ERREUR : Employe-e inexistant-e ...

=====

Menu : Gestion du personnel

=====

1) Recruter un nouveau/une nouvelle employe-e

2) Consulter les informations d'un-e employe-e

-----

3) Imprimer les etats

-----

4) Quitter l'application

-----

Choix -> 3

=====

Sous menu : 3) Imprimer les etats

=====

- 1) Liste des employe-es par matricule
- 2) Liste des salaires par deux semaines
- 3) Liste des employe-es

-----

4) Quitter

-----

Choisir un etat -> 1

=====

Menu :        Gestion du personnel

=====

- 1) Recruter un nouveau/une nouvelle employe-e
- 2) Consulter les informations d'un-e employe-e

-----

3) Imprimer les etats

-----

4) Quitter l'application

-----

Choix -> 3

=====

Sous menu : 3) Imprimer les etats

=====

- 1) Liste des employe-es par matricule
- 2) Liste des salaires par deux semaines
- 3) Liste des employe-es

-----

4) Quitter

-----

Choisir un etat -> 2



```
=====
Menu : Gestion du personnel
=====
```

- ```
1) Recruter un nouveau/une nouvelle employe-e
2) Consulter les informations d'un-e employe-e
-----
3) Imprimer les etats
-----
4) Quitter l'application
-----
```

Choix -> 3

```
=====
Sous menu : 3) Imprimer les etats
=====
```

- ```
1) Liste des employe-es par matricule
2) Liste des salaires par deux semaines
3) Liste des employe-es

4) Quitter

```

Choisir un etat -> 3

```
=====
Menu : Gestion du personnel
=====
```

- ```
1) Recruter un nouveau/une nouvelle employe-e
2) Consulter les informations d'un-e employe-e
-----
3) Imprimer les etats
-----
4) Quitter l'application
-----
```

Choix ->