### 

### 8PRO128 Programmation orientée objet

###### Automne 2019

# Travail pratique #3 (10% de la session) (modifié 14 nov)

##### Date de remise

**À remettre d’ici le 21 28 novembre 2019 (jeudi)**

(Peut se faire seul ou en équipe de 2)

**Objectif général**

Ce travail a pour objectif d’introduire l’étudiant à la notion de polymorphisme en programmation orientée objet et sa traduction en langage C++.

**Notions du langage C++ à acquérir**

Déclaration et définition du polymorphisme. Classe virtuelle.

**INSTRUCTIONS:**

* Votre travail pratique sera remis sous forme de dossier compressé. Ce dossier pourrait ne contenir qu’un document Word (le rapport décrit plus bas) et un dossier de projets (ou une solution contenant des projets) Visual Studio.
* Pour chacune des classes, vous aurez à créer un fichier de déclaration (.h) et un fichier de définition (.cpp). Le nom de chacun de ces fichiers doit être significatif de son contenu. Exemple : « Point.cpp », « Point.h » ou autre. Mettre des commentaires dans votre programme (fichier, but, auteur, date ou autre). Décrire les variables au besoin. Vous remettrez également une copie d’écran montrant l’exécution de votre programme.
* Remettre un diagramme de classes*.*

### Partie 1

Transformez les classes ***Point*** et ***Segment*** du premier TP en classes ***Template***s afin qu’elles puissent contenir des points aux coordonnées de types différents (int, double, char, …).

### Partie 2

Ajoutez à votre TP2 la déclaration d’un tableau de figures et y insérer des objets de types différents ( Triangle, Rectangle, Carré, ou autre).

Exemple : Figure \* tabFig[12] ;

tabFig [0] = new Triangle();

tabFig [1] = new Rectangle();

tabFig [2] = new Carre();

etc.

for ( int i = 0; i < nbFig; i++ )

cout << tabFig[i];

### Partie 3

Modifiez le système de paie de la figure 10.1 (volume de Deitel) pour ajouter l’attribut **private** appelé **dateNaissance** (un objet de **Date** du chapitre 8 de Deitel) à la classe **Employe**. Présumez que la paie est traitée une fois par mois. Créez un tableau (**vector**, **array** ou autre…) de références d’**Employe** pour conserver les différents objets **Employe**. Ensuite, dans une boucle, votre programme calculera la rémunération pour chaque **Employe** (en utilisant le polymorphisme). De plus, ajoutez une prime de 100 $ au montant de la paie de la personne si le mois correspond à celui de l’anniversaire de naissance de l’**Employe**.

Il vous faudra sans doute modifier le constructeur de la classe **Employe** pour supporter le nouvel attribut **private** **dateNaissance**.

**Bon travail!**