



POLYTECHNIQUE
MONTRÉAL

Questionnaire Contrôle Périodique5

LOG3430

Sigle du cours

Identification de l'étudiant(e)		
Nom :	Prénom :	
Signature :	Matricule :	Groupe :

Sigle et titre du cours		Groupe	Trimestre
LOG3430 - Méthodes de test et de validation du logiciel		Tous	20171
Professeur		Local	Téléphone
Soumaya Medini		C-624	
Jour	Date	Durée	Heures
Mardi	28 Mars 2017	1 heure	

Documentation	Calculatrice	
<input type="checkbox"/> Aucune <input checked="" type="checkbox"/> Toute <input checked="" type="checkbox"/> Voir directives particulières	<input type="checkbox"/> Aucune <input checked="" type="checkbox"/> Toutes <input type="checkbox"/> Non programmable	Les cellulaires, agendas électroniques ou téléavertisseurs sont interdits.

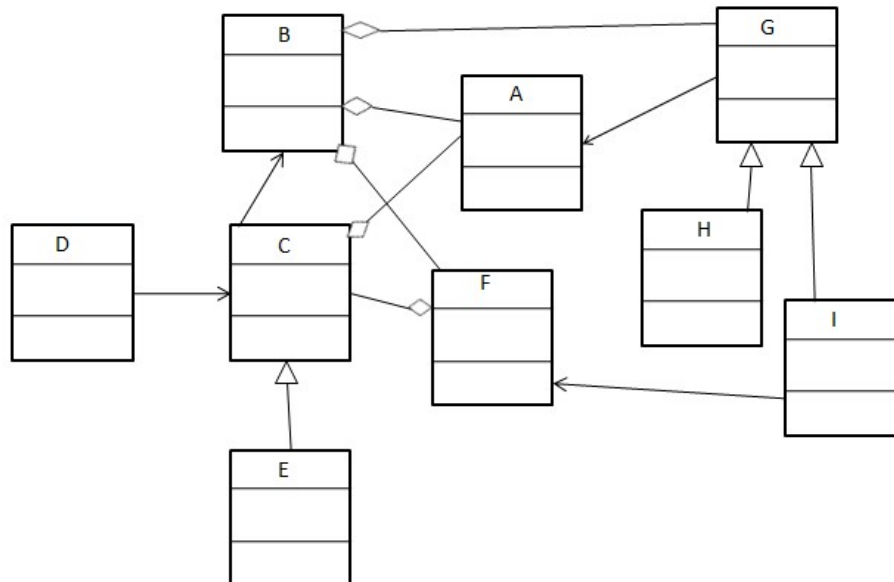
Directives particulières
Toute documentation est permise, ainsi que les calculatrices, à l'exception toutefois des téléphones cellulaires et de tout dispositif connecté à Internet.

Important	Cet examen contient <input type="text" value="2"/> exercices sur un total de <input type="text" value="6"/> pages (excluant cette page)
	La pondération de cet examen est de <input type="text" value="5"/> %
	Vous devez répondre sur : <input checked="" type="checkbox"/> le questionnaire <input type="checkbox"/> le cahier <input type="checkbox"/> les deux
	Vous devez remettre le questionnaire : <input checked="" type="checkbox"/> oui <input type="checkbox"/> non

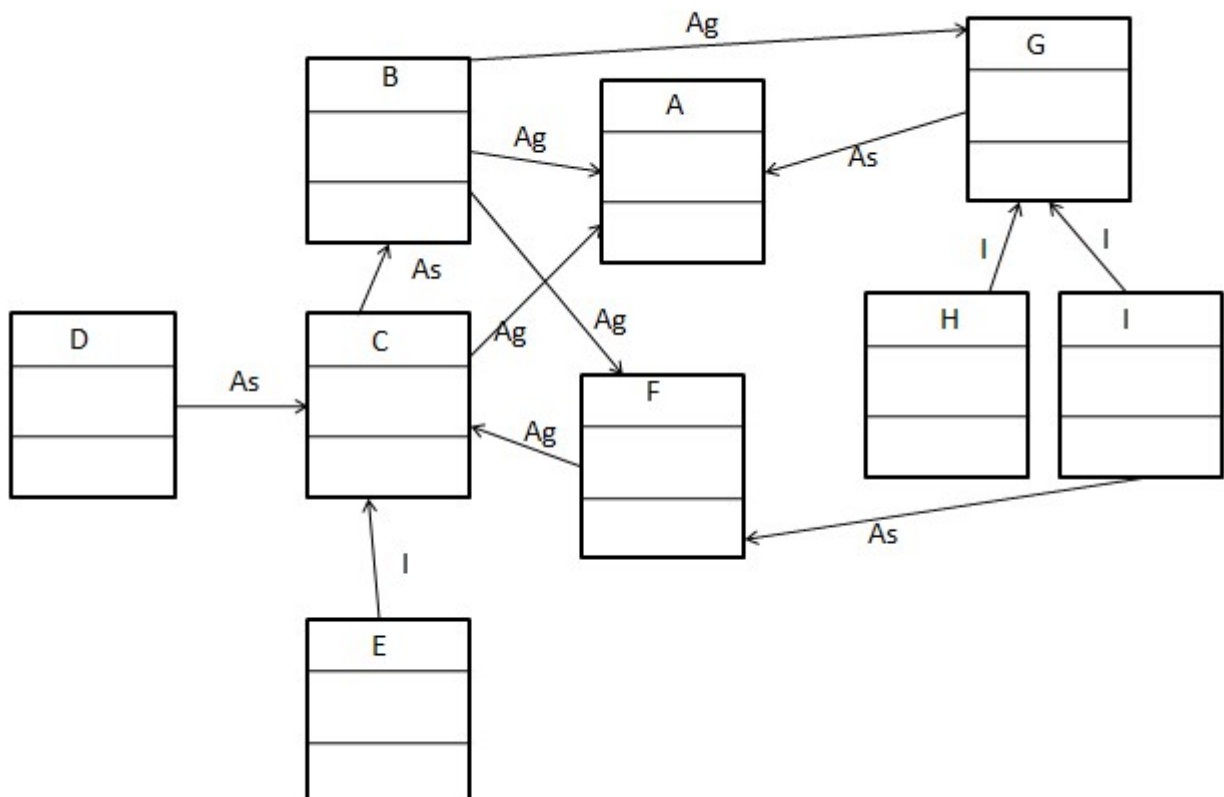
L'étudiant doit honorer l'engagement pris lors de la signature du code de conduite.

Exercice 1 – 10 points

Il faut calculer l'ordre d'intégration (en minimisant le nombre de stubs à développer) pour le diagramme de classe suivant :



Q1.1) Complétez le diagramme relation objet (ORD) correspondant au diagramme de classe: (2 points)



Q1.2) Considérez le diagramme ORD de Q1.1. Est-ce qu'il faut développer un ou plusieurs stubs?

(1 point)

- a. Non ☐
- b. Oui pour : __B_____ ☒



Q1.3) Considérez le diagramme ORD de Q1.1. Est-ce qu'il faut supprimer une ou plusieurs relations?

(1 point)

- a. Non ☐
- b. Oui la relation entre : __B-C_____ ☒

Q1.4) Pour chaque classe X, retrouvez CFW(X) soit l'ensemble des classes qui peuvent être affectées par un changement dans la classe X. Si vous pensez qu'un changement dans la classe X impacte la classe Y, alors mettez une croix à la case ligne X et colonne Y. Utiliser le diagramme ORD modifié si vous avez répondu oui à Q1.1 et/ou Q1.2. (2 points)

Classe impactée:

	A	B	C	D	E	F	G	H	I
A		X	X		X	X	X	X	X
B									
C				X	X	X			X
D									
E									
F		X							X
G		X						X	X
H									
I									

Changement dans:



Q1.5) Donnez l'ordre d'intégration en remplissant le tableau des niveaux de test. Si vous pensez qu'une classe X doit être testée au niveau i, alors mettez une croix à la case ligne i et colonne X. (2 points)

	A	B	C	D	E	F	G	H	I
1	X								
2			X				X		
3				X	X	X		X	
4		X							X
5									

Q1.6) Considérez que les classes C et G sont abstraites. Donner l'ordre d'intégration en remplissant le tableau des niveaux de test. Si vous pensez qu'une classe X doit être testée au niveau i, alors mettez une croix à la case ligne i et colonne X. (2 points)

	A	B	C	D	E	F	G	H	I
1	X								
2			X		X		X	X	
3				X		X			
4		X							X
5									

Exercice 2 – 10 points

Considérez le programme suivant. On suppose que le client, classe C, a déjà été testé et on doit tester le client avec le serveur, classe S.

```
public class C {
    private S as1, as2;

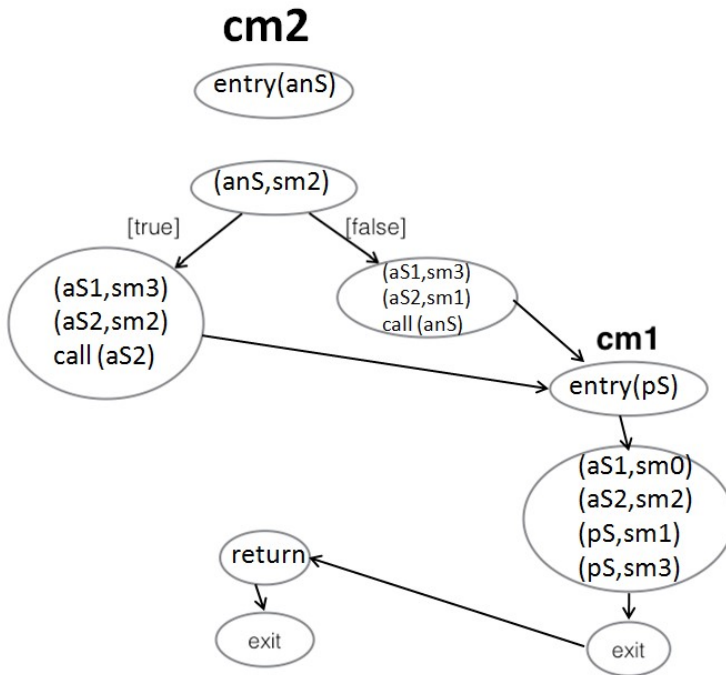
    public void cm2(final S anS) {
        if (anS.sm2(0)) {
            this.as1.sm3();
            this.as2.sm2(2);
            this.cm1(as2, true);
        } else {
            this.as1.sm3();
            this.as2.sm1();
        }
    }
}
```

```
        this.cm1(anS, true);
    }
}

private void cm1(final S pS, final boolean b) {
    this.aS1.sm0(b);
    this.aS2.sm2(2);
    pS.sm1();
    pS.sm3();
}
} //Class C
```

```
public class S {
public void sm0(final boolean b) {
    // ..some code
    return;
}
public boolean sm1() {
    // ..some code
    return false;
}
public int sm2(int q) {
    // ..some code
    return p;
}
public boolean sm3() {
    // ..some code
    return true;
}
} //Class S
```

Q2.1) Complétez l'ICFG annoté pour la méthode cm2: (3 points)



Q2.2) Pour les trois séquences de méthodes client fournies cm1, cm2, et cm1.cm2, identifiez les séquences de méthodes et instances du server déclenchées. Ne faites **aucune hypothèse sur l'égalité des instances**. (3 points)

cm1	(cm1,sm0,aS1) (cm1,sm2,aS2) (cm1,sm1.sm3,pS)
cm2	(cm2,sm2,anS) (cm2,sm2.sm1.sm3,anS) (cm2,sm3.sm0,aS1) (cm2,sm2.sm2.sm1.sm3,aS2) (cm2,sm1.sm2,aS2)
cm1.cm2	(cm1.cm2,sm1.sm3,pS) (cm1.cm2,sm2,anS) (cm1.cm2,sm2.sm1.sm3,anS) (cm1.cm2,sm0.sm3.sm0,aS1) (cm1.cm2,sm2.sm2.sm2.sm1.sm3,aS2) (cm1.cm2,sm2.sm1.sm2,aS2)

Q2.3) Pour les trois séquences de méthodes client fournies cm1, cm2, et cm1.cm2, identifiez les séquences de méthodes et instances du server déclenchées. Cette fois, **supposez que les séquences de méthodes client sont appelées sur l'attribut as1**. (2 points)

cm1	(cm1,sm0.sm1.sm3,aS1) (cm1,sm2,aS2)
cm2	(cm2,sm2.sm3.sm0,aS1) (cm2,sm2.sm3.sm0.sm1.sm3,aS1) (cm2,sm2.sm2.sm1.sm3,aS2) (cm2,sm1.sm2,aS2)
cm1.cm2	(cm1.cm3,sm0.sm1.sm3.sm2.sm3.sm0,aS1) (cm1.cm2,sm0.sm1.sm3.sm2.sm3.sm0.sm1.sm3,aS1) (cm1.cm2,sm2.sm2.sm2.sm1.sm3,aS2) (cm1.cm2,sm2.sm1.sm2,aS2)

Q2.4) Considérez votre réponse à Q2.3. Identifiez les séquences client/serveur à exécuter si on doit satisfaire le Critère 2 de redondance. (2 points)

cm1	(cm1,sm0.sm1.sm3,aS1) (cm1,sm2,aS2)
cm2	(cm2,sm2.sm3.sm0,aS1) (cm2,sm2.sm3.sm0.sm1.sm3,aS1) (cm2,sm2.sm2.sm1.sm3,aS2) (cm2,sm1.sm2,aS2)
cm1.cm2	(cm1.cm3,sm0.sm1.sm3.sm2.sm3.sm0,aS1) (cm1.cm2,sm0.sm1.sm3.sm2.sm3.sm0.sm1.sm3,aS1) (cm1.cm2,sm2.sm2.sm2.sm1.sm3,aS2) (cm1.cm2,sm2.sm1.sm2,aS2)