

**ÉPREUVE**

**Durée 1 heure 30 minutes**

*L'utilisation des documents est interdite*

**Nom & Prénom :** ..... **Groupe :** .....

**Exercice 01 : (6 Pts)** Cocher la bonne réponse : (bonne réponse : **0,5 Pts** / fausse réponse : **0 Pts**).

1. Les défauts sont introduits : - Tout au long du cycle du projet. <input checked="" type="checkbox"/> - Uniquement dès que l'on programme, les défauts ne sont essentiellement que les conséquences d'erreurs de programmation. <input type="checkbox"/> - Eventuellement en conséquence d'une spécification ambiguë. <input type="checkbox"/>
2. Les tests sont : - Inutiles la plupart du temps du fait de l'amélioration continue des techniques de développement. <input type="checkbox"/> - Indispensables pour réduire les coûts et les délais tout en augmentant la qualité. <input checked="" type="checkbox"/> - Importants mais impraticables donc à éliminer. <input type="checkbox"/>
3. Quelle affirmation définit le mieux à l'activité d'analyse : - Description de ce que le logiciel devra faire. <input checked="" type="checkbox"/> - Description de la solution proposée. <input type="checkbox"/> - Description des algorithmes de vérification des résultats. <input type="checkbox"/>
4. Qu'est-ce que UML ? - Une méthode de conception de logiciels. <input type="checkbox"/> - Un langage d'action et d'activité. <input type="checkbox"/> - Un langage de modélisation. <input checked="" type="checkbox"/>
5. Le cycle de vie en spirale est utilisé : - Pour des projets dont les enjeux sont importants. <input checked="" type="checkbox"/> - Pour des projets de taille et de complexité moyenne. <input type="checkbox"/> - Pour des projets de taille et de complexité grande. <input type="checkbox"/>
6. Le cycle de vie en V est utilisé : - Pour des projets dont les enjeux sont importants. <input type="checkbox"/> - Pour des projets de taille et de complexité moyenne. <input type="checkbox"/> - Pour des projets de taille et de complexité grande. <input checked="" type="checkbox"/>
7. Pour maîtriser la complexité des systèmes logiciels, il convient de : - Procéder selon une démarche bien définie. <input checked="" type="checkbox"/> - Se baser sur des principes et des méthodes. <input type="checkbox"/> - D'utiliser des outils performants. <input type="checkbox"/>
8. Les principales causes de la crise du logiciel sont : - Coût du changement et de maintenance. <input type="checkbox"/> - Mauvaise compréhension des besoins et des fonctionnalités du système. <input checked="" type="checkbox"/> - Coût du développement et du matériel. <input type="checkbox"/>
9. Pour qu'un logiciel soit extensible et réutilisable, il faut qu'il soit découpé en modules : - Faiblement couplés. <input checked="" type="checkbox"/> - Fortement couplés. <input type="checkbox"/>
10. Pour qu'un logiciel soit extensible et réutilisable, il faut qu'il soit découpé en modules : - A forte cohésion. <input checked="" type="checkbox"/> - A faible cohésion. <input type="checkbox"/>
11. Un logiciel est efficace s'il utilise : - Les ressources d'une manière optimale. <input checked="" type="checkbox"/> - Des algorithmes simples et efficaces. <input type="checkbox"/>
12. La fiabilité d'un logiciel est l'aptitude d'un logiciel à fonctionner : - Dans des conditions anormales. <input checked="" type="checkbox"/> - Sous différents environnements matériels et logiciels. <input type="checkbox"/>

**Exercice 02 : (3 Pts)**

1/ Définir l'axe de chacun des diagrammes UML suivants : (chaque bonne réponse : **0,5 Pts**)

Diagramme de classe	Diagramme de séquence	Diagramme de cas d'utilisations	Diagramme d'objets
Statique	Dynamique	Fonctionnel	Statique

2/ Quelle est la différence entre les deux activités suivantes ? (**0,5 Pts – 0,5 Pts**)

Validation des logiciels	Vérification des logiciels
Construire le <b>bon</b> logiciel qui doit <b>répondre aux besoins des utilisateurs</b>	Construire le logiciel <b>correctement</b> .

**Exercice 03 : (3 Pts)** (chaque bonne réponse : **0,25 Pts**)

Citer ce qui suit :

3 Outils de tests	- Générateurs de données - Analyseurs dynamiques - Simulateurs
3 types de programmation	- Impérative - Déclarative / Logique - Orientée Objets
3 types de maintenance	- Perfective / Evolutive - Adaptative - Corrective
3 concepts de structuration	- La modularité - L'abstraction - Le masquage d'information - La hiérarchie

#### **Exercice 04 : (4 Pts)**

Un abonné d'une médiathèque fait une demande d'emprunt d'un film spécifique. Le gestionnaire vérifie à la fois l'historique de l'emprunt de son client (s'il est à jour) et la disponibilité du film demandé. L'enregistrement de l'emprunt ne peut avoir lieu que si les deux paramètres sont validés. Une fois l'enregistrement validé, le stock et le compte client sont mis à jour.

- Proposer un diagramme de flot de données.



### Exercice 05 : (4 Pts)

Une équipe d'informatique est composée de développeurs. Elle travaille pour une entreprise et possède un identifiant et un logo. Un développeur est caractérisé par le numéro de sa carte d'identité, son nom, son prénom et son email, il utilise un ordinateur qui lui est personnel. Un développeur peut être un programmeur spécialisé dans un ou plusieurs langages de programmation, ou un concepteur expert dans une méthode ou langage de modélisation. Les développeurs écrivent le code. Les concepteurs dialoguent avec les utilisateurs et créent les modèles.

- Présenter le diagramme de classes relatif à la description ci-dessus.

