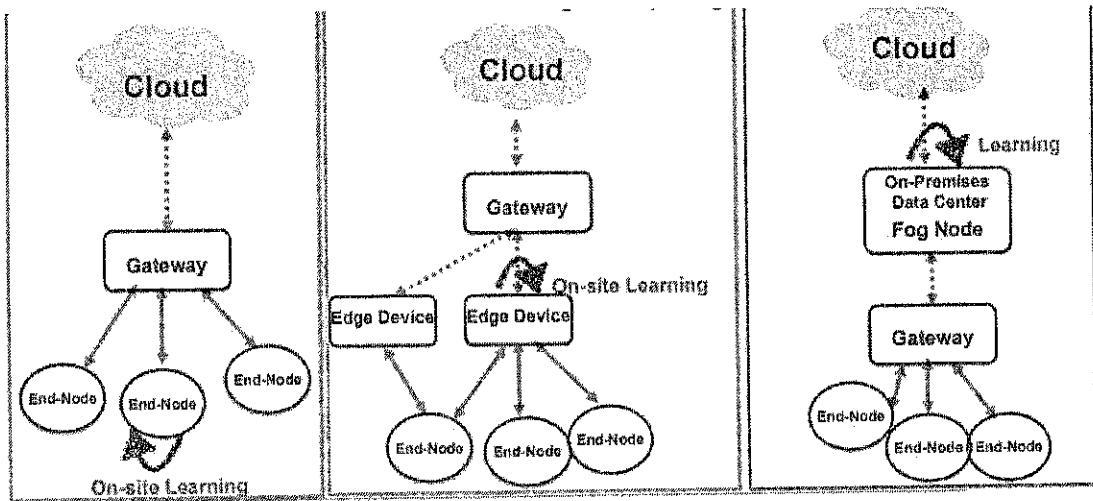


Exercice 3 :

Les solutions aux problématiques de l'IoT adoptent l'une des trois configurations ci-dessous. En argumentant à chaque fois par un exemple concret, donnez les avantages et inconvénients de chaque architecture.



Exercice 4 :

Dans le protocole VANET expliquer la différence entre les modes suivants :

- Geographical unicast (Geounicast)
- Topologically-Scoped Broadcast (TSB)
- Geographical Broadcast (Geobroadcast)
- Geographical Anycast (Geoanycast)

Exercice 5 :

- 1/ Expliquer par un schéma l'architecture SDN (Software Defined Networking) en explicitant les rôles respectifs de chaque composant.
- 2/ Quelle est la relation entre le NFV et le SDN ?
- 3/ Quelle est la différence entre le Inter flow network coding et le Intra flow network coding.
Donner une solution de chaque type.

**Examen Principal – S2 – 2023/2024**

Classe : 2^{ème} Ing GL	Matière : Internet of Things	Enseignant : Zied Jaoua
Date : 22 / 5 / 2024	Documents autorisés : Non	
Durée de l'examen : 1h30	Nombre de pages : 02	

Exercice 1 :

1. Quels sont les problèmes à résoudre dans les réseaux VANET qu'on ne trouve pas dans ceux MANET ?
2. Pourquoi a-t-on recours à une métrique particulière dans le routage des réseaux 802.11s ? Quels sont ses paramètres essentiels ?
3. Dans la norme 802.11s trois modes de routage peuvent être utilisés indiquer les différences entre eux et leurs avantages respectifs.
4. Comparer entre les réseaux Mesh et Ad hoc du point de vue : Qualité de service, routage et mobilité.

Exercice 2 :

Expliquez le code suivant et donner son utilité lors de l'utilisation de FreeRTOS.

```
static const uint16_t timer_divider = 80;
static const uint64_t timer_max_count = 1000000;
static const int adc_pin = A0;
static hw_timer_t *timer = NULL;
static volatile uint16_t val;
static SemaphoreHandle_t bin_sem = NULL;
void IRAM_ATTR onTimer() {
    BaseType_t task_woken = pdFALSE;
    val = analogRead(adc_pin);
    xSemaphoreGiveFromISR(bin_sem, &task_woken);
    if (task_woken) {
        portYIELD_FROM_ISR();
    }
}
```

- Dans le fichier « **movies.dat** », l'année d'apparition de chaque film est indiquée entre parenthèses.
- Dans le fichier « **users.dat** », l'âge est décrit par une tranche comme suit :

★ 1: "moins de 18"
★ 18: "18-24"
★ 25: "25-34"
★ 35: "35-44"
★ 45: "45-49"
★ 50: "50-55"
★ 56: "56+"

- Dans le fichier « **users.dat** », Occupation décrit la profession comme suit :

★ 0: "other" or not specified
★ 1: "academic/educator"
★ 2: "artist"
★ 3: "clerical/admin"
★ 4: "college/grad student"
★ 5: "customer service"
★

2. Codes

Vous disposez du code Python suivant pour établir un contexte Spark et définir la fonction **ParseMovies()**.

```
from pyspark import SparkContext
from pyspark.sql import SparkSession

spark = SparkSession\
    .builder\
    .master('local[*]')\
    .getOrCreate()
sc = spark.sparkContext

def parseMovies(row):
    splitted = row.split("::")
    return (int(splitted[0]), splitted[1], splitted[2])
```

Annexe

1. Aperçu des fichiers

```

1::Toy Story (1995)::Animation|Children's|Comedy
2::Jumanji (1995)::Adventure|Children's|Fantasy
3::Grumpier Old Men (1995)::Comedy|Romance
4::Waiting to Exhale (1995)::Comedy|Drama
5::Father of the Bride Part II (1995)::Comedy
6::Heat (1995)::Action|Crime|Thriller
7::Sabrina (1995)::Comedy|Romance
8::Tom and Huck (1995)::Adventure|Children's
9::Sudden Death (1995)::Action

```

Figure 1 : Extrait de « movies.dat » (MovieID::Title::Genres)

```

1::531::4::978302149
1::3114::4::978302174
1::608::4::978301398
1::1246::4::978302091
2::1357::5::978298709
2::3068::4::978299000
2::1537::4::978299620
2::647::3::978299351
2::2194::4::978299297
2::648::4::978299913

```

Figure 2 : Extrait de « ratings.dat » (UserID::MovieID::Rating:: Timestamp)

```

1::F::1::10::48067
2::M::56::16::70072
3::M::25::15::55117
4::M::45::7::02460
5::M::25::20::55455
6::F::50::9::55117
7::M::35::1::06810
8::M::25::12::11413
9::M::25::17::61614
10::F::35::1::95370
11::F::25::1::04093

```

Figure 3 : Extrait de « users.dat » : (UserID::Gender::Age::Occupation::Zip-code)

EXAMEN TYPE B

~~HPC-GT~~ Ingé - Infor

1. DEVOPS n'est pas un :
 - a) Ensemble de techniques et d'outils facilitant le passage du développement à la production.
 - b) Modèle de fonctionnement de l'entreprise
 - c) Modèle d'interactions entre les équipes
 - d) Modèle de test des logiciels
2. Laquelle des affirmations suivantes est incorrecte ?
 - a) DevOps ne convient qu'aux start-up.
 - b) DevOps convient aux produits et services logiciels dans l'industrie.
 - c) DevOps convient aux nouveaux produits et services logiciels.
 - d) Certaines des initiatives DevOps les plus exemplaires ont été lancées dans des entreprises dotées d'organisations informatiques géantes et matures.
3. Un merge est toute version ayant :
 - a) Degré sortant strictement supérieur à 1.
 - b) Degré entrant strictement supérieur à 1.
4. Un Commit stocke l'état d'une partie du dépôt à un instant donné. Il contient :
 - a) Une description du pointeur
 - b) Des informations sur la technologie du commit
 - c) Un pointeur vers un Tree (arbre racine) dont on souhaite sauver l'état.
 - d) L'état d'activité de l'auteur
5. Un tag est utilisé pour identifier l'état du dépôt au moment d'une release.
 - a) Vrai
 - b) Faux
6. Les commandes git sont de la forme:
 - a) git commande options
 - b) git options commande
 - c) commande git options
 - d) options commande git
7. Qu'est-ce qu'un commit ?
 - a) Une étape validée du code qui apparaît dans l'historique du dépôt.
 - b) L'un des multiples historiques contenus dans un dépôt git.
 - c) C. N'importe quelle modification récente faite sur un des fichiers du dépôt.
8. Comment connaître l'état courant d'un dépôt ?
 - a) git add
 - b) git checkout
 - c) git status
 - d) git branch
9. Qu'est-ce que HEAD ?
 - a) Le serveur où on pousse son code.
 - b) Un curseur pointant sur un commit qu'on peut déplacer avec git checkout.
 - c) Une interface pour utiliser git.
10. Git permet de :
 - a) Gérer plusieurs versions du code d'un logiciel/script
 - b) Corriger automatiquement du code
 - c) Explorer l'historique du code d'un logiciel
 - d) Obtenir de l'aide sur la syntaxe en Python
11. Une branche est :
 - a) Une nouvelle modification ajoutée à un dépôt
 - b) Une ligne d'historique du dépôt

EXAMEN TYPE B

- c) Une opération de fusion
12. Le Dockerfile est un fichier
- Java/python
 - c/c++
 - texte
13. Quelle est la principale différence entre une machine virtuelle (VM) et un conteneur ?
- Un conteneur est une boîte qui contient un logiciel Windows alors qu'une VM fonctionne généralement sous Linux.
 - Un conteneur permet de faire des applications distribuées dans le cloud contrairement aux machines virtuelles.
 - Un conteneur partage le noyau du système hôte alors qu'une machine virtuelle virtualise son propre noyau indépendant.
14. Quels sont les principaux atouts de Docker ?
- Il permet de rendre compatible tous les logiciels avec le cloud (AWS, etc.) et facilite l'IoT.
 - Il utilise le langage Go qui est de plus en plus populaire et accélère les logiciels qui l'utilise.
 - Il permet d'uniformiser les déploiements logiciels et facilite la construction d'application distribuées.
15. Pour créer un conteneur Docker à partir du code d'un logiciel il faut d'abord :
- Écrire un Dockerfile qui explique comment empaqueter le code puis construire l'image Docker avec docker build.
 - Créer un cluster avec docker-machine puis compiler le logiciel avec Docker Stack.
16. Un volume Docker est :
- Un espace de stockage connecté à un ou plusieurs conteneurs docker.
 - Une image fonctionnelle à partir de laquelle on crée des conteneurs identiques.
 - Un snapshot de l'application que l'on déploie dans un cluster comme Swarm.
17. Un Compose file ou fichier Compose permet :
- D'installer Docker facilement sur des VPS et de contrôler un cluster.
 - D'alléger les images et de détecter les failles de sécurité dans le packaging d'une application.
 - De décrire une application multiconteneurs, sa configuration réseau et son stockage.
18. Indiquez la ou les affirmations vraies : La philosophie de Docker est basée sur :
- L'immutabilité, c'est-à-dire le fait de jeter et recréer un conteneur pour le changer plutôt que d'aller modifier l'intérieur.
 - Le cloud, c'est-à-dire la vente de plateforme et de logiciel "as a service".
 - L'infrastructure-as-code, c'est-à-dire la description d'un état souhaité de l'infrastructure hébergeant l'application
19. Indiquez-la ou les affirmations vraies :
- Docker est très pratique pour distribuer un logiciel mais tous les conteneurs doivent obligatoirement être exposés à Internet.
 - Docker utilise un cloud pour distribuer facilement des logiciels dans de nombreuses versions.
 - Docker est une catastrophe en terme de sécurité car les conteneurs sont peu isolés.
20. Docker Swarm est :
- Un cloud où pousser et récupérer les images Docker de la terre entière.
 - Une solution de clustering et d'orchestration intégrée directement avec Docker.
 - Un logiciel qui complète ce qu'offre Kubernetes en y ajoutant des fonctionnalités manquantes
21. Quelle est la différence entre une image et un conteneur Docker?
- Une image est exécutable, tandis qu'un conteneur ne l'est pas
 - Une image est un modèle de base, tandis qu'un conteneur est une instance en cours d'exécution
 - Une image est utilisée pour stocker des données, tandis qu'un conteneur est utilisé pour stocker des fichiers
 - Il n'y a pas de différence entre une image et un conteneur

EXAMEN TYPE B

22. Quel est le concept central de Docker?
- a) Conteneurisation de processus
 - b) Virtualisation de machines
 - c) Gestion de configuration
 - d) Conteneurisation d'applications
23. Quelle est la commande utilisée pour exécuter un conteneur Docker?
- a) docker create
 - b) docker build
 - c) docker run
 - d) docker start
24. Qu'est-ce qu'un volume Docker?
- a) Une unité de mesure de la taille des conteneurs Docker
 - b) Un type de réseau permettant la communication entre conteneurs Docker
 - c) Un mécanisme permettant le stockage de données persistantes pour les conteneurs Docker
 - d) Une fonctionnalité permettant la création automatique de conteneurs Docker
25. Comment se fait la communication entre les conteneurs Docker?
- a) À travers le réseau Docker par défaut
 - b) À travers des câbles ethernet physiques
 - c) Par le transfert de fichiers entre systèmes de fichiers partagés
 - d) Il n'y a pas de communication possible entre les conteneurs Docker
26. DEVOPS n'est pas un :
- a) Ensemble de techniques et d'outils facilitant le passage du développement à la production.
 - b) Modèle de fonctionnement de l'entreprise
 - c) Modèle d'interactions entre les équipes
 - d) Modèle de test des logiciels
27. Comment mettre à jour une image Docker?
- a) Supprimer l'image et la recréer avec la nouvelle version
 - b) Télécharger la nouvelle version et la redéployer
 - c) Exécuter une commande spéciale pour mettre à jour l'image
 - d) Il n'est pas possible de mettre à jour une image Docker
28. Quelle est la principale différence entre Docker et Kubernetes?
- A. Docker est utilisé pour la virtualisation, tandis que Kubernetes est utilisé pour la conteneurisation
- B. Docker est un orchestrateur, tandis que Kubernetes est un moteur d'exécution
- C. Docker gère les conteneurs individuels, tandis que Kubernetes gère l'orchestration de plusieurs conteneurs
- D. Il n'y a aucune différence entre Docker et Kubernetes
29. Comment se fait la mise à l'échelle des conteneurs Docker?
- A. En augmentant la capacité mémoire des conteneurs
- B. En exécutant plusieurs instances des mêmes conteneurs
- C. En déléguant la gestion à un orchestrateur Docker
- D. Il n'est pas possible de mettre à l'échelle les conteneurs Docker
30. Quelle est la principale différence entre Docker et Kubernetes?
- A. Docker est utilisé pour la virtualisation, tandis que Kubernetes est utilisé pour la conteneurisation

EXAMEN TYPE B

- B. Docker est un orchestrateur, tandis que Kubernetes est un moteur d'exécution
- C. Docker gère les conteneurs individuels, tandis que Kubernetes gère l'orchestration de plusieurs conteneurs
- D. Il n'y a aucune différence entre Docker et Kubernetes

31. Quelle est la commande utilisée pour supprimer un conteneur Docker?

- A. docker delete
- B. docker stop
- C. docker kill
- D. docker rm

32. Quelle est la commande utilisée pour afficher la liste des conteneurs Docker en cours d'exécution?

- A. docker list
- B. docker show
- C. docker status
- D. docker ps

33. Qu'est-ce qu'un registre Docker?

- A. Un service permettant le stockage et la distribution des images Docker
- B. Un outil utilisé pour la gestion des couches de conteneurs Docker
- C. Un fichier de configuration pour déployer des clusters Docker
- D. Il n'y a pas de registre Docker

34. Quelle est la commande utilisée pour pousser une image vers un registre Docker?

- A. docker pull
- B. docker create
- C. docker tag
- D. docker push

35. Comment partager un volume entre un conteneur Docker et l'hôte?

- A. Il n'est pas possible de partager un volume entre un conteneur Docker et l'hôte
- B. À l'aide de l'option de montage (-v) lors de l'exécution du conteneur
- C. En utilisant une connexion réseau entre le conteneur et l'hôte
- D. En copiant manuellement les fichiers entre le conteneur et l'hôte

36. Quelle est la fonctionnalité principale de Docker Compose?

- A. Gérer les conteneurs individuels
- B. Créer des images Docker
- C. Déployer et gérer des applications multi-conteneurs

EXAMEN TYPE B

D. Gérer la mise à l'échelle des conteneurs Docker

37. Qu'est-ce qu'un réseau Docker?

A. Un service de messagerie pour les conteneurs Docker

B. Un mécanisme permettant la communication entre conteneurs Docker

C. Un outil pour surveiller les performances des conteneurs Docker

D. Il n'y a pas de réseau Docker

38. Comment mettre à jour une image Docker?

A. Supprimer l'image et la recréer avec la nouvelle version

B. Télécharger la nouvelle version et la redéployer

C. Exécuter une commande spéciale pour mettre à jour l'image

D. Il n'est pas possible de mettre à jour une image Docker

39. Quelles sont les deux composantes principales d'un conteneur Docker?

A. Un espace de noms et un groupe de processus

B. Un système de fichiers et des bibliothèques partagées

C. Un moteur d'exécution et des dépendances système

D. Il n'y a qu'une seule composante principale dans un conteneur Docker

40. Qu'est-ce qu'un conteneur Docker?

A. Un environnement virtuel isolé utilisé pour exécuter des applications

B. Une machine virtuelle légère sans système d'exploitation

C. Une instance en cours d'exécution d'une image Docker

D. Une fonctionnalité permettant l'émulation d'un environnement réseau virtuel

