

TSR – Segon Parcial

Aquest examen consta de 20 qüestions d'opció múltiple. En cada qüestió, només una opció és correcta. Has de contestar en una fulla a part. Cada resposta correcta aporta 0.5 punts a la qualificació. Cada resposta errònia descompta -0.167 punts.

TEORIA

1. En la configuració de desplegaments, aquest és un exemple d'“injecció de dependències”:

a	Generar una excepció quan el port a utilitzar ja estava assignat a un altre servidor (és a dir, a un diferent i ja en funcionament).
b	Iniciar una altra rèplica servidora quan la càrrega cresca; és a dir, gestionar accions d'escalat horitzontal.
c	Conèixer dinàmicament (o fins i tot actualitzar) quin és l'“endpoint” d'un altre component del servei que anem a utilitzar seguidament.
d	No desplegar cap instància d'algun component del servei.

2. Quina de les següents tasques NO ESTÀ relacionada amb el desplegament?

a	Instal·lació d'un component en certs ordinadors.
b	Depuració de l'aplicació.
c	Resolució de dependències.
d	Actualització dels programes a una nova versió.

3. Suposem que hem escrit un algorisme AL1 per a aconseguir consens en un sistema distribuït en el qual s'assumeix un model de fallades arbitràries (*bizantines*). Llavors:

a	AL1 és inútil perquè cap sistema real es comporta així (és a dir, el model de fallades arbitràries no es presenta en cap cas real).
b	AL1 haurà de replicar tots els seus processos utilitzant el model passiu perquè les fallades arbitràries només poden gestionar-se amb aquest model de replicació.
c	AL1 serà més fàcil de portar a un sistema real que un algorisme AL2 que assumisca fallades de parada, perquè és més difícil que un sistema complisca els supòsits d'AL2.
d	AL1 serà un dels algorismes més senzills perquè el model de fallades arbitràries facilita en gran manera la gestió de fallades en un sistema.

4. El model de replicació actiu...

a	...pot gestionar les fallades en les rèpliques més fàcilment que el model passiu.
b	...és incapaç de gestionar les fallades arbitràries.
c	...no pot proporcionar consistència seqüencial.
d	...és utilitzat per omissió en els conjunts de rèpliques de MongoDB.

TSR – Segon Parcial

5. Si es consideren les restriccions que imposa el teorema CAP, quan gestionem particions de la xarxa utilitzant el model de partició primària...

a	...es pot proporcionar consistència seqüencial en els subgrups de processos minoritaris.
b	...es pot proporcionar consistència seqüencial en aquell subgrup amb una majoria de processos.
c	...es garanteix la disponibilitat i la consistència entre rèpliques es relaxa.
d	...es renuncia a la disponibilitat en aquell subgrup amb una majoria de processos.

6. Si es consideren les restriccions que imposa el teorema CAP, quan hàgem de garantir la disponibilitat d'un servei desplegat en diversos centres de còmput suportant el particionat de la xarxa, llavors...

a	...podrem utilitzar consistència seqüencial entre totes les rèpliques del servei.
b	...haurem d'utilitzar un model de consistència relaxada; per exemple, la consistència FIFO.
c	...no hi haurà cap restricció que afecte a la consistència entre rèpliques.
d	...la disponibilitat del servei no es podrà garantir en un escenari com aquest, compost per diversos centres de còmput distribuïts geogràficament.

7. Quin d'aquests serveis és més elàstic?

a	Un que afeg una rèplica cada hora, independentment de la càrrega suportada a cada moment.
b	Un que reporta periòdicament la càrrega actual, la capacitat de servei i el nombre de rèpliques a l'administrador, qui decideix quina acció d'escalat ha d'aplicar-se.
c	Un que adapta el seu nombre de rèpliques a la càrrega actual de manera autònoma, mantenint una qualitat de servei adequada (segons el seu SLA).
d	Un que no monitoritza mai l'ús de recursos.

8. Quina d'aquestes aproximacions NO MILLORA l'escalabilitat d'un servei?

a	Distribució de tasques.
b	Distribució de dades.
c	La presa de decisions mitjançant votació entre tots els processos participants.
d	Usar components amb funcionalitat similar a la memòria cau.

TSR – Segon Parcial

SEMINARIS

9. Quina característica distingeix a les màquines virtuals dels contenidors Docker?

a	Algunes aplicacions a executar en elles poden utilitzar biblioteques que no estan presents en el sistema operatiu amfitrió.
b	La necessitat d'un sistema operatiu amfitrió.
c	L'aïllament entre contenidors i/o màquines virtuals.
d	Les màquines virtuals poden executar aplicacions que no podrien ser executades en el sistema operatiu amfitrió.

10. Amb l'ordre "docker commit" es pot...

a	Finalitzar una transacció iniciada per a descarregar o pujar una imatge Docker a un magatzem ("repository").
b	Crear una imatge Docker, prenent un Dockerfile com a base.
c	Iniciar un contenidor Docker, prenent una imatge Docker com a base.
d	Crear una imatge Docker, a partir d'un contenidor Docker.

11. Assumim el següent Dockerfile, l'ús del qual no generarà cap error durant el seu processament:

```
FROM zmq  
COPY ./myProgram.js /server.js  
EXPOSE 8000 8001  
CMD node /server.js
```

Quina de les següents afirmacions ÉS FALSA?

a	Existeix una imatge Docker "zmq", en el magatzem local o en el global.
b	Els components resultants tindran un procés "server" que pot escoltar en el port 8001 del contenidor. Aquest port correspon al port 8000 en l'amfitrió.
c	La imatge generada amb aquest Dockerfile podrà executar altres programes, a part de "node server.js".
d	Quan es genere una imatge amb aquest Dockerfile, ha d'haver-hi un fitxer anomenat "myProgram.js" en la mateixa carpeta que el fitxer Dockerfile.

TSR – Segon Parcial

12. Quina afirmació ÉS FALSA sobre l'ordre “docker-compose”?

a	S'assumeix que hi haurà un fitxer “docker-compose.yml” en la carpeta on s'executa l'ordre.
b	Sempre necessita que hi haja un Dockerfile per a cadascun dels components a executar.
c	Pot utilitzar-se per a controlar un servei basat en múltiples components. Les instàncies de tots els components s'executen en un mateix amfitrió.
d	La seua acció “scale” pot utilitzar-se per a modificar el nombre d'instàncies d'un component del servei.

13. Quina afirmació ÉS FALSA sobre els models de consistència “ràpids”?

a	Quan un procés executa una acció de lectura o escriptura, l'acció pot retornar el control sense haver intercanviat cap missatge amb altres processos.
b	El model de consistència FIFO és ràpid.
c	El model de consistència causal és ràpid.
d	El model de consistència seqüencial és ràpid.

14. Per a implantar el model de consistència FIFO (utilitzant ZeroMQ sobre TCP), es necessita:

a	Un procés seqüenciador.
b	Que cada procés difonga les seues escriptures amb un socket PUB i reba les escriptures dels altres utilitzant un únic socket SUB.
c	Que cada procés difonga les seues escriptures amb múltiples sockets PUB (un per variable) i un únic socket SUB per a rebre les escriptures dels altres processos.
d	Utilitzar rellotges vectorials i afegir el rellotge vectorial actual a cada valor escrit quan siga difós als altres processos.

15. El mòdul “cluster” en NodeJS...

a	...permet que un conjunt de contenidors siga desplegat sobre diversos amfitrions.
b	...amplia les ordres “docker-compose” per a desplegar programes NodeJS més ràpid que sense el mòdul “cluster”.
c	...permet que els programes NodeJS siguin multi-fil.
d	...permet que un conjunt de processos treballadors NodeJS compartisquen els ports assignats a un procés <i>master</i> NodeJS.

TSR – Segon Parcial

16. Estem escrivint un programa NodeJS amb el mòdul “cluster”. Utilitza tants treballadors com processadors hi haja. El procés mestre informará l'usuari cada segon sobre quantes peticions han sigut ateses fins al moment pels treballadors.

```
var cluster = require('cluster');
var http = require('http');
if (cluster.isMaster) {
    var numReqs = 0;
    setInterval(function() { console.log("numReqs =", numReqs); }, 1000);
    function messageHandler(msg) {
        numReqs++;
    }
    var numCPUs = require('os').cpus().length;
    for (var i=0; i < numCPUs; i++) cluster.fork();
    /* (1) Worker message management instructions should be here. */
    /* (2) Worker regeneration code should be here, if any.      */
} else {
    http.Server(function(req, res) {
        res.writeHead(200); res.end('hello world\n');
        process.send({ cmd: 'notify' });
    }).listen(8000);
}
```

Tria quines instruccions necessita el procés mestre per a gestionar (1) els missatges dels treballadors:

a	<code>cluster.on('message', messageHandler);</code>
b	<code>for (var i in cluster.workers) cluster.workers[i].on('message', messageHandler);</code>
c	<code>for (var i in workers) workers[i].on('notify', messageHandler);</code>
d	<code>cluster.on('notify', messageHandler);</code>

17. En el programa anterior desitgem mantenir el nombre de treballadors constant. Per a fer això, si un mor haurà de ser reemplaçat per un nou treballador. Quines instruccions (2) es necessiten per a implantar aquesta funcionalitat?

a	<code>cluster.on('exit', function(worker, code, signal) { cluster.fork(); });</code>
b	<code>cluster.on('exit', function(worker, code, signal) { worker.restart(); });</code>
c	<code>cluster.on('exit', function(worker, code, signal) { worker.process.fork(); });</code>
d	<code>cluster.on('exit', function(worker, code, signal) { worker.process.start(); });</code>

TSR – Segon Parcial

18. MongoDB utilitza els següents mecanismes d'escalabilitat:

a	Replicació activa (o de màquina d'estats) i distribució de tasques.
b	Replicació passiva i distribució de dades (particionat horitzontal).
c	Replicació activa i distribució de dades (particionat horitzontal).
d	MapReduce.

19. Quina afirmació ÉS FALSA sobre els processos “mongos”?

a	Redirigeixen les peticions que reben.
b	Mantenen una memòria cau de les metadades de configuració de <i>sharding</i> de MongoDB.
c	Estan situats en els ordinadors clients.
d	Cadascun pot ser reemplaçat per un “conjunt de rèpliques”.

20. Quina afirmació ÉS FALSA sobre el “write concern” de MongoDB?

a	Especifica quantes rèpliques han de persistir les modificacions causades per una operació d'escriptura, esborrat o modificació.
b	Valors baixos en el “write concern” redueixen el temps de servei de les escriptures, però posen en perill la persistència de les dades en cas de fallada.
c	El seu valor recomanat és “majority” perquè proporciona un bon compromís entre rendiment i tolerància a fallades.
d	Necessita una confirmació per part del client per a cada operació sol·licitada.