This exam has a maximum mark of 10 points and consists of 32 questions. Each question has 4 choices and only one of them is correct. Each correct answer provides 10/32 points, and each error discounts 10/96 points. You should deliver the answers sheet.

These programs have been developed to generate a system. We refer subsequently to that system as ABC:

```
// client.js
const zmq = require("zeromq")
const url = process.argv[2]
let push = zmq.socket('push')
push.connect(url)
push.send('hello 1')
push.send('hello 2')
push.send('hello 3')
```

```
// a.js
const zmq = require("zeromq")
const [port1,port2] = process.argv.slice(2)
let pull = zmq.socket('pull')
let push = zmq.socket('push')
pull.bindSync('tcp://*:'+port1)
push.bindSync('tcp://*:'+port2)
pull.on('message', (m) => {push.send(m)})
```

```
// b.js
const zmq = require("zeromq")
const [url1,url2] = process.argv.slice(2)
let pull = zmq.socket('pull')
let push = zmq.socket('push')
pull.connect(url1)
push.connect(url2)
pull.on('message', (m) => {push.send(m)})
```

```
// c.js
const zmq = require("zeromq")
const port = process.argv[2]
let pull = zmq.socket('pull')
pull.bindSync('tcp://*:'+port)
pull.on('message', (m) => {console.log(m+"")})
```

ABC has been manually deployed, using the following commands (each line in a different terminal)

```
node c 8002
```

```
node a 8000 8001
```

```
node b tcp://127.0.0.1:8001 tcp://127.0.0.1:8002
```

```
node b tcp://127.0.0.1:8001 tcp://127.0.0.1:8002
```

```
node client tcp://127.0.0.1:8000
```

This output is shown in the first terminal (that of command: node c 8002):

```
hello 1
hello 2
hello 3
```

That deployment should be automated using containers so that clients access ABC remotely through port 80 of the host. To this end, a tsr2021/ubuntu-zmq base image has been used (with the ubuntu distribution and support for Node.js and ZeroMQ) and the following directory structure:

```
p2/docker-compose.yml
p2/a/Dockerfile
p2/a/a.js
p2/b/Dockerfile
p2/b/b.js
p2/c/Dockerfile
p2/c/c.js
```

The Dockerfile for a.js has a line with this content:

```
CMD node a 8000 8001
```

The Dockerfile for c.js has, among others, this instruction:

```
CMD node c 8002
```

You should also consider that THE CLIENT IS NOT AN ELEMENT IN THE AUTOMATED DEPLOYMENT of the ABC system.

1 *In the manual deployment of the ABC system:*

- **a** There are 5 distinct components, with an instance each one.
- **b** There are 4 distinct components, one of them with 2 instances.
- **c** There is a single component, since we deploy all programs in the same machine.
- **d** There is a single component, since all programs use the same execution support (Node.js + ZeroMQ).

2 *In the ABC system WITH AUTOMATED DEPLOYMENT:*

- **a** There are no dependences.
- **b** There are only dependences of clients on 'a'.
- **c** There are dependences of 'b' on 'a' and 'c', and of clients on 'a'.
- **d** There are only dependences of 'b' on 'a' and 'c'.

3 *Regarding the automation of the ABC deployment in order to run 'a', 'b' and 'c' in the same host:*

- **a** The use of containers is not suitable, since the deployment of multiple elements in a single host must use virtual machines.
- **b** Elements 'b' and 'c' may use containers, but 'a' has to handle remote client messages and, therefore, it may only be deployed using a virtual machine.
- **c** Elements 'a', 'b' and 'c' may use containers.
- **d** If we run 'a', 'b' and 'c' in the same machine, we cannot use remote clients: clients must run in that same host, too.

4 *The file p2/b/Dockerfile in the ABC system has the following content*

```
FROM tsr2021/ubuntu-zmq
COPY ./b.js b.js
CMD node b ...
```

Choose the appropriate contents for the ellipsis of its third line:

- **a** `tcp://127.0.0.1:8001 tcp://127.0.0.1:8002`
- **b** `tcp://127.0.0.1:8001 $URL`
- **c** `$URL tcp://127.0.0.1:8002`
- **d** `$URL1 $URL2`

5 *Let us assume that file p2/b/Dockerfile of the ABC system has a correct content and that our current working directory is p2/. What is the command that creates the 'b' image (with name 'imb')?*

- **a** `docker build -t imb .`
- **b** `docker commit -t imb .`
- **c** `docker build -t imb ./b`
- **d** `docker run -t imb .`

6 *Let us assume that the ellipsis represents, if needed, the list of arguments or environment variable values to use by program 'b' of the ABC system and that image 'imb' (associated to 'b') has been created correctly. Choose the command that executes an instance of 'b' in a container:*

- **a** `docker build imb ...`
- **b** `docker commit imb ...`
- **c** `docker start imb ...`
- **d** `docker run imb ...`

7 *Which of these lines is convenient to include in file p2/a/Dockerfile of the ABC system?*

- **a** `EXPOSE 8000 8001`
- **b** `PORTS 8000 8001`
- **c** Both EXPOSE 8000 8001 and PORTS 8000 8001
- **d** Neither EXPOSE 8000 80001 or PORTS 8000 8001

8 *In regard to service 'a' in the docker-compose.yml file of the ABC system:*

- **a** It requires both 'links:' and 'environment:' clauses.
- **b** It needs the 'links:' clause, but no 'environment:'.
- **c** It needs the 'environment:' clause, but no 'links:'.
- **d** It needs no 'environment:' or 'links:' clause.

9 *In regard to service 'a' in the docker-compose.yml file of the ABC system:*

- **a** It requires a 'ports:' clause.
- **b** All other choices are incorrect.
- **c** It needs the 'expose:' clause, but no 'ports:'.
- **d** It needs no 'expose:' or 'ports:' clause.

10 *In file docker-compose.yml of the ABC system:*

- **a** It only needs an 'environment:' clause in service 'b'.
- **b** It only needs an 'environment:' clause in services 'a' and 'c'.
- **c** It only needs an 'environment:' clause in service 'a'.
- **d** No service needs an 'environment:' clause.

11 *Let us assume that in system ABC all files are correct. A valid command to run one instance of 'a', one of 'b' and one of 'c' is*

- **a** `docker run ima & docker run imb & docker run imc &`
- **b** `docker-compose up`
- **c** `docker build --scale ima=1 --scale imb=1 --scale imc=1`
- **d** That goal cannot achieved with a single command: some parts should be manually done (find out the container IP, edit the Dockerfiles, etc.)

12 *If system ABC (i.e., elements 'a', 'b' and 'c') has been automatically deployed, the correct way to run a client in the same machine is:*

- **a** Execute in another terminal the command `node client tcp://127.0.0.1:80`
- **b** Execute in another terminal the command `node client tcp://127.0.0.1:8000`
- **c** None. The client cannot be started in the same machine (it must run in another host)
- **d** Execute the command `docker run client tcp://127.0.0.1:80`

13 *In regard to the interoperability of sockets DEALER and REP, which of the following sentences is FALSE?*

- **a** All messages sent by a DEALER to a REP have to include, at least, one empty segment.
- **b** The DEALER has to operate synchronously since REP sockets demand this.
- **c** Several DEALERs can be connected and operating with several REPs.
- **d** A REP can be connected and may operate with a set of sockets in which there are several DEALERs and REQs.

14 *Let us modify the clients-broker-workers system described in theory (see, for example, the programs in Unit 4) in the following way:*

1) The broker, instead of two ROUTER sockets has only one ROUTER that interacts with both clients and workers.

2) Consequently, clients and workers have been modified so that they share the same endpoint (i.e., the ROUTER) and they are anonymous, that is, they have no value explicitly assigned to their identity attribute.

Choose the correct statement:

a The code of the original broker can be transformed (with minor modifications) to obtain that of this new broker so that the resulting system operates in the usual way.

b This modification cannot be carried out, since that new broker will not distinguish clients from workers.

c This modification only makes sense if workers use REP sockets.

d All other choices are incorrect.

15 *Choose the FALSE statement:*

a Messages sent by DEALER sockets always have to include some empty segment.

b ROUTER sockets are asynchronous.

c ROUTER sockets use identities to handle sent or received messages.

d DEALER sockets can interoperate among them.

16 *Choose the FALSE statement:*

a DEALER sockets are synchronous.

b DEALER sockets enqueue the messages that have not been able to send yet.

c DEALER sockets enqueue the received messages that haven't been delivered yet.

d DEALER sockets may interoperate with ROUTER sockets.

17 *The following trace, where…*

- an element 1xW2 means that process (1) applies on variable (x) the write (W) operation generating value (2)

- an element 3yR1 means that process (3) applies on variable (y) the read (R) operation obtaining value (1)

```
1xW1 2xR1 4xR1 3xR1 2xW2 1xW3 3xR3 4xR2 3xR2 4xR3
```

a Fulfils strict consistency

b Fulfils sequential consistency

c Fulfils causal consistency

d All other choices are false

18 *If a given trace fulfils sequential consistency*

a we can not ensure that it fulfils processor consistency

b we can not ensure that it fulfils causal consistency

c we can not ensure that it fulfils cache consistency

d All other choices are false

19 *In a system with non-deterministic operations and without Byzantine errors, and with an average operation length 10 times the cost to propagate and apply updates*

a We must use active replication if we deploy a single service

b We must use passive replication only if we deploy more than one service

c Both active and passive replication models may be used

d We must use passive replication independently on the number of deployed services

20 *In a system with deterministic operations and possibility of Byzantine errors, and with an average operation length 10 times the cost to propagate and apply updates*

- **a** We must use passive replication
- **b** We must use multi-master replication
- **c** We must use active replication exclusively if we deploy a single service and passive replication if we deploy more than one service
- **d** We must use active replication independently of the number of deployed services

21 *What happens if the passive model allows that backup replicas directly process and respond the read-only operations sent by clients?*

- **a** That service tolerates arbitrary failures.
- **b** That service improves its tolerance to network partitions.
- **c** Non-deterministic operations will not be managed in a consistent way.
- **d** That service improves its throughput.

22 *A NoSQL datastore is a database in which (among other possible characteristics):*

- **a** The relational model and the transactional support are not used.
- **b** The database schema is simplified.
- **c** Data partitioning and distribution among different nodes are allowed.
- **d** All other choices are correct.

23 *According to the CAP theorem, when a network partition happens…*

- **a** All subgroups continue answering to their clients and replicated services provide sequential consistency.
- **b** Replicated services may respect causal consistency if they remain available.
- **c** A primary partition model is used and all service replicas may answer to their clients.
- **d** No activity will be permitted in all service replicas while the network remains partitioned.

24 *Choose the TRUE statement about multi-master replication:*

- **a** It manages more failure models than active replication.
- **b** It needs a total-order broadcast to guarantee its inter-replica consistency.
- **c** It allows that the instructions of each requested operation are run in a single (but possibly different) replica.
- **d** It requires that read-only operations are answered by all its replicas to support the arbitrary failure model.

25 *Which is the MongoDB replication type?*

- **a** None, since it improves its scalability by means of data partitioning/distribution.
- **b** Multi-master.
- **c** Passive if the network is not partitioned, and multi-master when there are partitions in the network.
- **d** Passive, but it admits in its configuration that back-up replicas respond to read-only requests.

26 *The Node.js cluster module of Node.js has this purpose:*

a Deploy a set of Node.js programs in a cluster of computers.

b Manage multiple threads of execution in a Node.js process.

c Manage a set of Node.js processes so that they can share the program to be run and the port in which they listen to requests.

d Facilitate the development of proxies that propagate requests to a cluster of computers where multiple servers may be deployed.

27 *In the development of Lab 3, we have created several images with the same name associated to distinct versions of the program. To avoid conflicts, we…:*

a …place those images with the same name in distinct directories.

b …remove previous images before generating a new one.

c There is no possibility of conflict, because new images automatically overwrite those pre-existing ones with the same name.

d There is no possibility of conflict, because the system chooses always the most recent version.

28 *In the CBW variant that accepts job types in Lab 3, we assume that:*

- the client has a single program that generates a job type (request) or another according to a command-line argument

- the worker has a single program that enables it for a job type or another according to a command-line argument

In that scenario, and assuming that later we want to control the number of instances of each type of client and worker:

a We may have the clients program in a single directory, and that of workers in a single directory

b We need a distinct directory per worker type, but for clients a single directory is enough

c We need a distinct directory per client type, but for workers a single directory is enough

d We need a distinct directory per client type, and a distinct directory per worker type

29 *In the CBW variant that accepts job types in Lab 3, we assume that:*

- the client has a single program that generates a job type (request) or another according to a command-line argument

- the worker has a single program that enables it for a job type or another according to a command-line argument

In that scenario, and assuming that later we want to control the number of instances of each type of client and worker:

a The file docker-compose.yml has to include a distinct entry per client type and a distinct entry per worker type

b The file docker-compose.yml has to include a distinct entry per client type, but a single entry for all workers is enough

c The file docker-compose.yml has to include a distinct entry per worker type, but a single entry for all clients is enough

d In file docker-compose.yml, we only need a single entry for all clients and a single entry for all workers

30 *The section of manual deployment of Lab 3 has this Dockerfile for clients*

```
FROM tsr2021/ubuntu-zmq
COPY ./2client.js myclient.js
CMD node myclient NEED_BROKER_URL
```

With imclient as its corresponding image. Let us consider this other Dockerfile

```
FROM imclient
ENTRYPOINT ["node"]
CMD ["myclient"]
```

placed in a new directory, where we run:

```
docker build -t nuevocliente .
```

In those conditions:

a The new Dockerfile is wrong.

b The image nuevocliente is not adequately built.

c With nuevocliente, we may build and run clients that can interoperate with the broker if they receive the suitable information.

d Although we may start new client containers with nuevocliente, they will not have any way to interoperate with the broker.

31 *The second part of Lab 3 considers the use of a logger in the conditions presented there. Consider the following variant: the logger only receives annotations from the broker and in this new deployment the first service to be deployed is the broker. To this end, we have modified adequately the broker and logger programs, only in the minimum required places, to take into account these conditions according to what is shown in this docker-compose.yml file*

```
version: '2'
services:
...
  bro:
    image: broker
    build: ./broker/
    expose:
      - "9997"
      - "9998"
      - "9999"
 log:
    image: logger
    build: ./logger/
    links:
      - bro
    volumes:
      # /tmp/logger.log DIRECTORY
must
      # exist on host and be
writeable
      - /tmp/logger.log:/tmp/cbwlog
    environment:
      - LOGGER_DIR=/tmp/cbwlog
      # The logger does connect
      - BROKER_URL=tcp://bro:9997
```

In these conditions:

a This deployment is not feasible and will report an error.

b The system may be deployed but when the broker starts first, the logger may not receive the first annotations.

c The system thus deployed will work adequately.

d In order to work correctly, the logger must be deployed before clients and workers.

32 *In the third session of Lab 3, we must deploy a worcli component that intercommunicates two client-broker-worker systems. Choose the correct sentence:*

a Worcli must be deployed in the host of the first broker.

b Worcli must be deployed in the host of the second broker.

c Worcli can only communicate with other Worcli.

d Worcli is not strictly necessary. The brokers of these two systems, without modifications, may communicate directly without any intermediate component.

NIST **Second Partial** **7 Jan 2022** **A**

Fill and deliver this answer sheet. Each question has a single correct answer. Don't forget to correctly write your personal data.

Don't fix a wrong answer with another mark: erase it or cover it with Tipp-Ex

A question with more than an answer will be discarded

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

DNI: _____

Surname: _____

First Name: _____

1	A	B	C	D	**21**	A	B	C	D
2	A	B	C	D	**22**	A	B	C	D
3	A	B	C	D	**23**	A	B	C	D
4	A	B	C	D	**24**	A	B	C	D
5	A	B	C	D	**25**	A	B	C	D
6	A	B	C	D	**26**	A	B	C	D
7	A	B	C	D	**27**	A	B	C	D
8	A	B	C	D	**28**	A	B	C	D
9	A	B	C	D	**29**	A	B	C	D
10	A	B	C	D	**30**	A	B	C	D
11	A	B	C	D	**31**	A	B	C	D
12	A	B	C	D	**32**	A	B	C	D
13	A	B	C	D					
14	A	B	C	D					
15	A	B	C	D					
16	A	B	C	D					
17	A	B	C	D					
18	A	B	C	D					
19	A	B	C	D					
20	A	B	C	D					