**Instructions**
On the desk you are only allowed to have:

- this exam,
- black/blue ink pens, and
- your DNI (or UPC identification).

**Electronic devices and backpacks are not allowed near the desk or your person.**
*Make sure to write your name/surname/DNI on every page (except this one).*

This exam consists of **4 open questions**. The maximum score $G$ attainable in this exam is 10. The grade of this exam is

$$\max\{G, 50\%G + 50\%E\}\ ,$$

where $E$ is your grade of the continuous assessment. If you don't submit this exam, your grade is $E$.

**21:00 h**: this exam ends (except for students with adaptations).

**Name:** ______________ **Surname(s):** ______________________________ **DNI:** ______________

Please use **only** the space below (or the back of the page) for your solution of the following exercise. Use blue or black ink. You can write your solution either in Catalan, Spanish, or English.

**1** (3 points) For each of the following languages, decide whether it is regular or not. In either case you should prove your claim. In case it is regular, give the minimum DFA recognizing the language and prove that it is indeed minimum.

(a) (1.6 points) For any $n \geq 1$, define

$$L_n = \{w \in \{0,1\}^* \mid \exists k \geq 0 \ \ \text{value}_2(w) = (2^n)^k\}.$$

That is, $L_n$ is the language containing the strings over the alphabet $\{0,1\}$ whose binary value is a power of $2^n$.

**Solution:** Define the DFA $A_n = (\{0,\dots,n{+}1\}, \{0,1\}, \delta, 0, \{1\})$, where the transition function is defined as $\delta(0,0) = 0$, $\delta(0,1) = 1$, $\delta(i,0) = i+1$ for all $i$ such that $1 \leq i < n$, and $\delta(n,0) = 1$; any other transition is directed to the sink state $n+1$. Since $A_n$ has a loop traversing the states $1,\dots,n$, the only strings accepted by $A_n$ are those of the form $10^{nk}$ for some $k$. Therefore, $L(A_n) = L_n$. To see that $A_n$ is minimum, consider the set of strings

$$S = \{\lambda, 1, 10, \dots 10^n\}.$$

For any two strings $x$ and $y$ in $S$, we can find a distinguishing extension for $L_n$, that is, a string $z$ s.t. $xz \in L_n \not\Leftrightarrow yz \in L_n$. Since all strings in $S$ have different lengths, suppose that $|x| < |y|$. If $x = \lambda$, then $z = 1$ is a dintinguishing extension. Otherwise, $x = 10^i$ and $y = 10^j$ for some $i, j$, with $i < j$. In this case, $z = 0^{n-i}$ is indeed a distinguishing extension. Then, $S$ is $L_n$-distinguishable and we know that any DFA for $L_n$ requires at least $|S| = n+2$ states. As a consequence, our DFA $A_n$ is minimum.

(b) (1.4 points) For any $n \geq 1$, define

$$L'_n = \{ww \in \{0,1\}^* \mid \exists k \geq 0 \ \ \text{value}_2(w) = (2^n)^k\}.$$

That is, $L'_n$ is the language containing the strings $ww$ over the alphabet $\{0,1\}$ such that the binary value of $w$ is a power of $2^n$.

**Solution:** We will see that the infinite set $S' = \{10^i \mid i \geq 0\}$ is $L'_n$-distinguishable. Take any two strings $x = 10^i$ and $y = 10^j$, $i < j$, in $S'$, and let $r$ be an integer such that $n$ divides $i + r$. Now, define the extension $z = 0^r 10^{i+r}$. Now we have that the string $xz$ is in $L'_n$ while $yz$ is not since $yz = 10^{j+r}10^{i+r}$ and, even if $j + r$ was a multiple of $n$, $yz$ is not of the form $ww$ required for $L'_n$. The existence of an infinite $L'_n$-distinguishable set implies that $L'_n$ cannot be regular.

**Name:** ______________ **Surname(s):** ________________________________ **DNI:** ______________

Please use **only** the space below (or the back of the page) for your solution of the following exercise. Use blue or black ink. You can write your solution either in Catalan, Spanish, or English.

**2** (2 points) Prove the following facts about context-free languages. CFL is the set of all context-free languages, DCFL is the set of all *deterministic* context-free languages. Recall that $\{a^n b^n c^n \mid n \in \mathbb{N}\} \notin \mathsf{CFL}$ and DCFL is closed under complement and inverse homomorphism.

(a) (1 point) Show that the language

$$L = \{0^i 1^j 2^k \mid i \neq j \vee j \neq k\} \in \mathsf{CFL}\ .$$

Construct a context-free grammar generating $L$ or a pushdown automata accepting $L$.

**Solution:** Notice that

$$L = L_1 \cdot \mathscr{L}(2^*) \cup L_2 \cdot \mathscr{L}(2^*) \cup \mathscr{L}(0^*) \cdot L_3 \cup \mathscr{L}(0^*) \cdot L_4\ ,$$

where $L_1 = \{0^i 1^j \mid i > j\}$, $L_2 = \{0^i 1^j \mid i < j\}$, $L_3 = \{1^j 2^k \mid j > k\}$, and $L_4 = \{1^j 2^k \mid j < k\}$.
A grammar generating $L_1$ is

$$\begin{cases} S_1 & \to & 0S_1 1 \mid A \\ A & \to & 0A \mid 0\ . \end{cases}$$

Similarly, grammars for $L_2$, $L_3$ and $L_4$ are respectively:

$$\begin{cases} S_2 & \to & 0S_2 1 \mid B \\ B & \to & 1B \mid 1 \end{cases} \qquad \begin{cases} S_3 & \to & 1S_3 2 \mid B \\ B & \to & 1B \mid 1 \end{cases} \qquad \begin{cases} S_4 & \to & 1S_4 2 \mid C \\ C & \to & 2C \mid 2\ . \end{cases}$$

Therefore a context-free grammar generating $L$ is the following:

$$\begin{cases} S & \to & S_1 \mid S_2 \mid S_3 \mid S_4 \mid S_1 C \mid S_2 C \mid AS_3 \mid AS_4 \\ S_1 & \to & 0S_1 1 \mid A \\ S_2 & \to & 0S_2 1 \mid B \\ S_3 & \to & 1S_3 2 \mid B \\ S_4 & \to & 1S_4 2 \mid C \\ A & \to & 0A \mid 0 \\ B & \to & 1B \mid 1 \\ C & \to & 2C \mid 2\ . \end{cases}$$

(b) (1 point) Show that $L \notin \mathsf{DCFL}$.

**Solution:** Since DCFL is closed under complement it is enough to show that $\overline{L} \notin \mathsf{DCFL}$. Since $\mathsf{DCFL} \subseteq \mathsf{CFL}$ it is enough to show that $\overline{L} \notin \mathsf{CFL}$.
Towards a contradiction suppose $\overline{L} \in \mathsf{CFL}$. Since CFL is closed w.r.t. intersection with a regular language we have that $\overline{L} \cap \mathscr{L}(0^*1^*2^*) = \{0^n 1^n 2^n \mid n \in \mathbb{N}\} \in \mathsf{CFL}$. Since CFL is closed under homomorphisms, then using the homomorphism $\sigma(0) = a$, $\sigma(1) = b$, $\sigma(2) = c$ we get that $\sigma(\overline{L} \cap \mathscr{L}(0^*1^*2^*)) = \{a^n b^n c^n \mid n \in \mathbb{N}\} \in \mathsf{CFL}$. Which contradicts the known fact that $\{a^n b^n c^n \mid n \in \mathbb{N}\} \notin \mathsf{CFL}$.

**Name:** ________________ **Surname(s):** ______________________________________ **DNI:** ________________

Please use **only** the space below (or the back of the page) for your solution of the following exercise. Use blue or black ink. You can write your solution either in Catalan, Spanish, or English.

**3** (3 points)

(a) (1.4 points) Prove that a language $A$ is in $\mathbf{R}$ (that is, it is decidable) if and only if there are two total computable functions $f, g : \mathbb{N} \to \mathbb{N}$ s.t. for each $x \in \mathbb{N}$

$$x \in A \iff f(x) > g(x) \; .$$

[*0.7 points are awarded for the "if" part of the exercise and 0.7 for the "only if" part.*]

**Solution:** ( $\Longrightarrow$ ) If $A \in \mathbf{R}$, then the characteristic function of $A$, $\chi_A$ is computable (and total). Recall that $\chi_A(x) = \begin{cases} 1 & \text{if } x \in A \, , \\ 0 & \text{if } x \notin A \, . \end{cases}$ We can then take $f(x) = \chi_A(x)$ and $g(x) = 0$. If $x \in A$ we have $f(x) = 1 > 0 = g(x)$, and for $x \notin A$, $f(x) = 0 = g(x)$. Therefore the functions $f$ and $g$ have the properties we wanted.

($\Longleftarrow$) To show that $A \in \mathbb{R}$ we construct a TM $M$ (given as pseudocode) that decides $A$:

```
input x {
    compute y = f(x) − g(x)
    if (y > 0) then
        accept
    else
        reject
}
```

Since $f$ and $g$ are total functions, $M$ always halts. If $x \in A$, then by assumption $f(x) - g(x) > 0$ and hence $y > 0$ and we enter the `if` condition where we accept. If $x \notin A$, by assumption $f(x) = g(x)$ or $f(x) < g(x)$. In either case it is not true that $y > 0$, and hence we enter the `else` case and reject.

(b) (1.6 points) A *useless* state in a Turing machine is a state that is never entered on any input string. Let $L = \{\langle p, Q \rangle \mid Q \text{ is a useless state in } M_p\}$, where $M_p$ is a Turing machine with Gödel number $p$.

Classify $L$ as belonging to $\mathbf{R}$ (decidible), $\mathbf{RE} \setminus \mathbf{R}$ (semi-decidable but not decidable), $\mathbf{coRE} \setminus \mathbf{R}$, or ouside both $\mathbf{RE}$ and $\mathbf{coRE}$. Justify your answer.

**Solution:** $L \in \mathbf{coRE} \setminus \mathbf{R}$. An alternative description of $L$ is the following

$$L = \{\langle p, Q \rangle \mid \forall x \, \forall t \, M_p(x) \text{ at execution step } t \text{ is not is state } Q\} \; .$$

The language of tuples $\langle p, Q, x, t \rangle$ s.t. $M_p(x)$ at execution step $t$ is not is state $Q$ is in $\mathbf{R}$: simply simulate $M_p$ on input $x$ for $t$ steps and check wheter you end-up in state $Q$ or not. This implies $L \in \mathbf{coRE}$. To show that $L \notin \mathbf{R}$ we construct a reduction from $\overline{K} = \{x \mid M_x(x)\uparrow\}$ to $L$: the reduction maps $x$ to $(p, Q)$, where $p$ is the Gödel number of the following TM.

```
input y {
    simulate M_x(x)
    go to state Q
}
```

The reduction is clearly a total computable function and for $x \in \overline{K}$, the machine $M_p$ never terminates simulating $M_x(x)$, hence $Q$ is useless, that is $(p, Q) \in L$. If $x \notin \overline{K}$, then the machine $M_p$ enters the state $Q$ (on every input $y$) and hence it is not useless. That is $(p, Q) \notin L$.

**Name:** ______ **Surname(s):** ______ **DNI:** ______

Please use **only** the space below (or the back of the page) for your solution of the following exercise. Use blue or black ink. You can write your solution either in Catalan, Spanish, or English.

**4** (2 points) A *strong* nondeterministic Turing machine is one that has three possible outcomes: **yes**, **no**, and **maybe**. We say that such a machine decides $L$ if this is true: If $x \in L$, then all computations end up with **yes** or **maybe**, and at least one with **yes**. If $x \notin L$, then all computations end up with **no** or **maybe**, and at least one with **no**.

Show the following implications.

(a) (1 point) If $L$ is decided by a strong nondeterministic Turing machine in polynomial time, then $L \in \mathsf{NP} \cap \mathsf{coNP}$.

**Solution:** Let $L$ be decided by a strong nondeterministic Turing machine (NTM) $S$ in polynomial time. Then, we can define the following NTM $M$ to recognize $L$: on an input $x$, simulate $S$ with input $x$ and if any computation branch ends up with **yes**, accept on that branch; if it ends up with **maybe** or **no**, reject. Clearly, $M$ accepts $x$ if and only if some branch accepts, which happens if and only if $S$ accepts $x$, that is, if $x \in L$. A similar machine $N$ can be defined to recognize $\overline{L}$ where **yes** and **no** are exchanged in the simulation. Now, $M$ proves that $L \in \mathsf{NP}$ and $N$ proves that $L \in \mathsf{coNP}$.

(b) (1 point) If $L \in \mathsf{NP} \cap \mathsf{coNP}$, then $L$ is decided by a strong nondeterministic Turing machine in polynomial time.

**Solution:** Let $M$ (resp., $N$) be a polynomial-time NTM for $L$ (resp., $\overline{L}$). We can combine $M$ and $N$ in order to define a strong NTM $S$ that decides $L$ in polynomial time: on input $x$, simulate $M(x)$ and $N(x)$ consecutively. If both simulations end up with a rejection, output **maybe**. Otherwise, if $M$'s simulation accepts, output **yes** and if $N$'s simulation accepts, output **no**. In case that $x \in L$, $M(x)$ must have some accepting path, which corresponds to an output **yes** for $S$; symmetrically, if $x \notin L$, $N(x)$ must have some accepting path and, then, $S$ will have some output **no**. It is also clear that **yes** and **no** cannot be present in the same computation tree of $S$ on any input and. Therefore, $S$ correctly decides $L$.