Lea cuidadosamente las instrucciones del Certamen Online Práctico en la siguiente página.

Instrucciones Certamen Online Práctico (COP) - INF-285/ILI-285
SCT - Vi.05.06.20 - v0.4

El presente documento tiene como finalidad clarificar qué esperar al momento de realizar un Certamen Online Práctico (COP). Además incluye sugerencias para que pueda ejecutarse con éxito.

1. Las preguntas de los COPs se componen de 2 partes: la parte teórica (✍) y la parte práctica o computacional (⌨), las cuales se destacarán con los símbolos indicados.

2. El tiempo estimado por pregunta es de aproximadamente 60 minutos. Luego de desarrollar su COP deberá:
   - Digitalizar su desarrollo teórico (✍), es decir convertirlo en PDF. Debe estar todo el desarrollo teórico en 1 solo documento, que puede contener varias páginas si es que fuera necesario. Su desarrollo debe estar ordenado.
   - Subir el PDF generado y su Jupyter Notebook de la componente práctica (⌨).
   - La componente práctica de todas las preguntas debe ser adjuntada en el único Jupyter Notebook entregado.

3. El formato de entrega del desarrollo teórico (✍) es `ROL.PDF` y de la componente práctica (⌨) es `ROL.ipynb`.

4. Es muy importante que prepare con anticipación la forma en que digitalizará su desarrollo teórico (✍). Por ejemplo se presenta la siguiente lista a modo de sugerencia:
   - CamScanner: Applicación disponible para Android y iPhone, https://www.camscanner.com.
   - LibreOffice: En la aplicación *Writer* (similar a Microsoft Word) uno puede importar imágenes y luego exportar el documento a PDF. Esta opción requiere que las imágenes sean traspasadas al computador primero. https://www.libreoffice.org.
   - Google Docs: Aplicación online, también similar a Microsoft Word, que permite importar imágenes y luego en "File", "Download" y PDF, se puede exportar en PDF. Esta opción requiere que las imágenes sean traspasadas al computador primero. https://www.google.com/drive/
   - Microsoft Word: Software de pago, permite importar imágenes y luego en "Guardar Como" se selecciona la opción PDF. Esta opción requiere que las imágenes sean traspasadas al computador primero. https://www.microsoft.com/en-us/microsoft-365/word
   - Cualquier otra aplicación que usted pueda conocer. No hay problema si las aplicaciones utilizadas incluyen alguna marca de agua o logo, lo importante si es que no cubran sus desarrollos.

5. Al finalizar su evaluación, deberá adjuntar la siguiente *Declaración de Trabajo Individual* **escrita a mano**. Al inicio de la declaración dice "Juro o prometo", usted debe elegir una de las dos opciones, no es necesario incluir las dos. Se sugiere buscar la diferencia *online*.

Declaración de Trabajo Individual: *Juro o prometo que la totalidad del trabajo que he entregado en esta evaluación corresponde a mi trabajo individual, y es el fruto de mi estudio y esfuerzo. Además declaro que no he recibido ayuda externa ni he compartido de forma alguna mi trabajo o desarrollos.* Nombre, Rol, Firma y Fecha: ____

6. Se sugiere tener esta declaración a mano para poder transcribirla y subirla en el PDF del desarrollo teórico al finalizar el COP. **Es requerido entregar la "Declaración de Trabajo Individual" para poder evaluar su COP.**

7. Es muy importante que usted entregue sus desarrollos en los tiempos indicados, dado que la plataforma Aula estará configurada para aceptar documentos sólo en los tiempos definidos. Documentos no entregados en los tiempos publicados, no serán aceptados por otros medios o de forma posterior.

8. Recordar que es posible eliminar uno de los primeros 3 COPs, por si llegara a tener problemas fortuitos al momento de la entrega, existe esa opción.

9. Se recuerda tener presente el Reglamento de Derechos y Deberes de los Alumnos en Casa Central y Campus Santiago.

10. Si no se cumplen estas instrucciones, su pregunta no será revisada o tendrá una penalización.

11. La última, pero no menos importante sugerencia, es que se asegure de entregar los archivos correctos, tanto del desarrollo teórico (✍) y el Jupyter Notebook correcto del desarrollo práctico (⌨).

<u>**Instrucciones**</u>: ***Usted tiene que mostrar todo su trabajo de forma clara y ordenada para obtener todos los puntos***. *Este certamen consta de 3 preguntas, las cuales serán entregadas todas juntas en un PDF. Sus desarrollos y código sebe ser subido a la plataforma Aula en los tiempos indicados. Puntos parciales serán entregados a preguntas incompletas. Respuestas finales sin desarrollo o* ***sin nombre*** *reciben 0 puntos.* Copy-and-Paste *de algoritmos reciben 0 puntos. ¡Éxito!*

Se recuerda que:

- Al finalizar la totalidad de su evaluación, deberá adjuntar la siguiente *Declaración de Trabajo Individual* **escrita a mano** en PDF de las preguntas teóricas.

Declaración de Trabajo Individual: *Juro o prometo que la totalidad del trabajo que he entregado en esta evaluación corresponde a mi trabajo individual, y es el fruto de mi estudio y esfuerzo. Además declaro que no he recibido ayuda externa ni he compartido de forma alguna mi trabajo o desarrollos.* Nombre, Rol, Firma y Fecha: ____

- Dada la modalidad online del COP, no se podrán realizar consultas sobre las preguntas durante la ejecución de la evaluación.
- A modo excepcional, se entregará un bono de un factor 1.1 quienes entreguen su COP-2 antes de las 11:45am.
- Se aplicará un descuento lineal si la entrega se produce posterior a las 18:00 hrs. El descuento será de 0 % a las 18:00 hrs y de un 100 % a las 20:00 hrs. Posterior a ese tiempo no será posible entregar sus respuestas. Esto se aplica al tiempo de entrega del último archivo subido a Aula.
- # **Avanzar a las siguientes páginas para ver las preguntas**.

1. Considere el siguiente sistema de ecuaciones lineales:

$$
\underbrace{\begin{pmatrix}
0 & \dots & \dots & \dots & 0 & b & a \\
0 & \dots & \dots & 0 & b & a & c \\
0 & \dots & 0 & b & a & c & 0 \\
\vdots & \iddots & \iddots & \iddots & \iddots & \iddots & \vdots \\
0 & \iddots & \iddots & \iddots & \iddots & \dots & \vdots \\
b & a & c & 0 & \dots & \dots & 0 \\
a & c & 0 & \dots & \dots & \dots & 0
\end{pmatrix}}_{A}
\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix}
=
\begin{pmatrix} 1 \\ 1/2 \\ 1/3 \\ 1/4 \\ 1/5 \\ \vdots \\ 1/(n-1) \\ 1/n \end{pmatrix}
\tag{1}
$$

donde se conoce que $|c| < |b| < \dfrac{|a|}{2}$ y $A \in \mathbb{R}^{n\times n}$.

(a) ✍, **[14 puntos]** Proponga un algoritmo **nuevo** basado en los algoritmos vistos en el curso que tome ventaja de la estructura y de la información entregada del sistema de ecuaciones lineales (1) que permita obtener los coeficientes x_1, x_2, $\dots$, y x_n, dado a, b, c y n. *Hint: You have complete freedom in how to find x_1, x_2, $\dots$, and x_n! Make sure you propose an algorithm that does find them. Hint2: You may consider aditional parameters, but you should justify their use.*

(b) ⌨, **[7 puntos]** Implemente el algoritmo propuesto considerando que el **input** es: a, b, c, y n; y el **output** es un NumPy Array con los coeficientes x_1, x_2, $\dots$, y x_n.

(c) ⌨, **[7 puntos]** Obtenga los coeficientes x_1, x_2, $\dots$, y x_n para el siguiente input: $a = 10$, $b = -4$, $c = 1/2$ y $n = 3000$.

(d) ⌨, **[7 puntos]** Implemente un algoritmo que dado los parámetros $a = 10$, $b = -4$, $c = 1/2$ y $n = 3000$ pueda determinar si unos coeficientes x_1, x_2, $\dots$, y x_n, satisfacen la ecuación matricial (1), y utilícelo para verificar si en (1c) obtuvo los coeficientes esperados.

2. Una agencia de seguridad necesita hacerle llegar de manera confiable un código a un espía para que pueda decodificar un mensaje y así conocer en donde se encuentra su siguiente objetivo. La agencia le enviará al espía una matriz $F \in \mathbb{R}^{M \times N^2}$ donde cada fila de F corresponderá a una matriz $A^{(i)} \in \mathbb{R}^{N \times N}$ vectorizada respecto a sus filas. Por ejemplo, si el espía recibe la siguiente matriz:

$$F = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 \\ b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 & b_8 & b_9 \\ c_1 & c_2 & c_3 & c_4 & c_5 & c_6 & c_7 & c_8 & c_9 \\ d_1 & d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 \\ e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 \\ f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_8 & f_9 \end{bmatrix}$$

Entonces la matriz $A^{(1)}$ es:

$$A^{(1)} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix}$$

Para decodificar el código el espía deberá identificar y etiquetar las matrices en una de las siguientes categorías:

a) Matrices simétricas, diagonal dominante y definidas positivas deben ser etiquetadas con un número 1.

b) Matrices simétricas, **no** diagonal dominante y definidas positivas deben ser etiquetas con un número 2.

c) El resto de las matrices tienen que ser etiquetadas con un número 3.

Por lo tanto, si en la matriz F de ejemplo todas las matrices $A^{(i)}$ son simétricas y definidas positivas entonces el código debiese ser 222222.

Restricción sólo válida para esta pregunta: *No utilizar funciones provenientes de librerías para obtener descomposiciones de matrices, por ejemplo PALU, Cholesky, SVD, etc, si es que fuera necesario. Sí está permitido reutilizar código visto en clases.*

Considerando lo anterior responda las siguientes preguntas:

(a) ✍, **[10 puntos]** Considerando que el espía no tiene acceso a los valores propios de las matrices $A^{(i)}$. Proponga un algoritmo que le permita identificar matrices simétricas, diagonal dominante y definidas positivas sin recurrir al cálculo de valores propios.

(b) ✍, **[10 puntos]** Considerando que el espía no tiene acceso a los valores propios de las matrices $A^{(i)}$. Proponga un algoritmo que le permita identificar matrices simétricas y definidas positivas sin recurrir al cálculo de valores propios. *Hint: Recall that Cholesky only works with symmetric and positive definite matrices.*

(c) ⌨, **[15 puntos]** Considerando sus dos respuestas anteriores implemente un algoritmo que reciba como **input** la matriz F en un Numpy Array (ver Jupyter Notebook entregado) y devuelva como **output** el código de decodificación en un NumPy Array.

3. Considere las siguientes definiciones.

   - La matriz de permutación J es una matriz similar a la identidad pero con unos en su diagonal secundaria. A continuación se presenta una matriz $J \in \mathbb{R}^{5\times 5}$ como ejemplo:

$$J = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

   - Una matriz $B \in \mathbb{R}^{n\times n}$ es bisimétrica si cumple que $B^T = B$ y $BJ = JB$, dicho de otra manera, es una matriz cuadrada que es simétrica con respecto a su diagonal principal y secundaria. Por ejemplo la siguiente matriz $B \in \mathbb{R}^{5\times 5}$ es bisimétrica:

$$B = \begin{bmatrix} a & b & c & d & e \\ b & f & g & h & d \\ c & g & i & g & c \\ d & h & g & f & b \\ e & d & c & b & a \end{bmatrix}.$$

   - Una matriz Z es una matriz cuyas entradas cumplen que $z_{i,j} \leq 0$, $i \neq j$. Es decir, es una matriz con coeficientes negativos o 0, excepto en la diagonal principal que sus valores son $z_{i,i} \geq 0$.

   Comentario previo a la realización de sus preguntas: *Las funciones para generar las matrices que se requieren en las preguntas se incluyen en el Jupyter Notebook base del certamen.*

   Ahora, resuelva las siguientes preguntas:

   (a) ✍, **[5 puntos]** Sea A una matriz Z y además bisimétrica. Proponga un algoritmo que resuelva el problema $A\mathbf{x} = \mathbf{b}$ para este tipo de matrices. Considere que A es no singular.

   (b) ⌨, **[5 puntos]** Implemente el algoritmo propuesto en el ítem (3a). El **input** de su algoritmo debe ser A, $\mathbf{b}$, y como **output** debe retornar $\mathbf{x}$.

   (c) ✍, **[5 puntos]** Sea A una matriz Z, bisimétrica y además cumple que $a_{i,i} > -\displaystyle\sum_{i\neq j}^{n} a_{i,j}$, $i \in \{1, 2, \ldots, n\}$. Proponga un algoritmo iterativo que resuelva el problema $A\mathbf{x} = \mathbf{b}$ para este tipo de matrices. Considere que el criterio de detención de su algoritmo es $\frac{\|\mathbf{r}_k\|}{\|\mathbf{b}\|} \leq \text{tol}$ o un máximo número de iteraciones de 5000, donde $\mathbf{r}_k$ es el vector residual en la iteración k y $\text{tol} = 10^{-8}$.

   (d) ⌨, **[5 puntos]** Implemente el algoritmo propuesto en el ítem (3c). El **input** de su algoritmo debe ser A, $\mathbf{b}$ y sus parámetros respectivos. El **output** debe ser $\mathbf{x}$.

   (e) ⌨, **[10 puntos]** Sea C una matriz Z y bisimétrica. Ahora, considere la siguiente matriz $A = C + \delta I \in \mathbb{R}^{n\times n}$, con $\delta \geq 0$. La matriz A cumple que $a_{i,i} > -\displaystyle\sum_{i\neq j}^{n} a_{i,j}$, $i \in \{1, 2, \ldots, n\}$. Estime el menor valor de $\delta \in [0, 100000]$ tal que su implementación del algoritmo en (3d) obtenga un tiempo de ejecución menor o igual que la implementación del algoritmo en (3b) para los tamaños de matrices $n \in \{100, 500, 1000, 2000\}$.

   (f) ✍, **[10 puntos]** ¿Qué puede concluir del experimento numérico anterior?