

ENSAYO 2

Instrucciones

- Antes de comenzar, lea atentamente las instrucciones, los requerimientos de entrega y las preguntas de la evaluación.
- La evaluación es de carácter individual y está sujeta a las normas del código de honor de la asignatura. Cualquier indicio de intervención de otra persona o de cualquier otro acto sancionable como falta al código de honor significará la calificación mínima al promedio de Teoría.
- En caso de detectar programas iguales o con suficientes similitudes entre sí, tanto en una misma sección como en secciones distintas, se citará a los casos para revisión. Con este fin, se compararán las entregas con un algoritmo de reconocimiento de texto y se revisarán los casos sospechosos mediante un proceso de validación visual que realizará el cuerpo docente del curso. En caso de que se determine que hubo copia, se sancionará aplicando la calificación mínima al promedio de Teoría.
- Estudiantes que entreguen fuera de plazo serán calificados con nota mínima en esta evaluación.
- Entregas que se realicen por vías distintas a Campus Virtual serán calificadas con nota mínima.
- En caso de que la plataforma presente un problema, puede enviar su archivo como respaldo al correo de contacto del profesor de Teoría, y posteriormente subir el archivo a Campus Virtual.
- Estudiantes que entreguen sin seguir los criterios de identificación estipulados en el apartado de «entrega» serán calificados con nota mínima en esta evaluación.
- La difusión de este enunciado solicitando ayuda, ofreciendo dinero a cambio de la resolución e incluso la publicación de éste dentro del plazo de desarrollo de la tarea, se considera un acto deshonesto, que daña la imagen de la Facultad de Ingeniería y de la Universidad de Santiago de Chile, y significará la reprobación inmediata de la asignatura y la aplicación de los procedimientos disciplinarios que el reglamento de ética de la Universidad dispone para estos casos.
- Considere que cualquier supuesto que el estudiante haga debe ser explicitado en los comentarios de código.
- La subida del archivo es responsabilidad de su autor, por lo que archivos que no estén en el formato estipulado, que vengan corruptos o con problemas para ser leídos no serán revisados.
- La evaluación (E2) se calificará considerando 100% puntaje de casos de prueba.

- Considera que los casos de prueba son solo referenciales y que tus respuestas se probarán y calificarán con casos distintos, los que siempre se comportarán de acuerdo a la especificación de cada ejercicio.
- Cualquier situación no contemplada en este documento será dirimida por la Facultad de Ingeniería.

Entrega

Para cada pregunta, cree un archivo `.py` con su RUN. Agregue al encabezado de su programa los siguientes datos de identificación. Considere que, de no agregarlos, el puntaje de su pregunta no contará.

```
# FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA
# FUNDAMENTOS DE COMPUTACIÓN Y PROGRAMACIÓN
# SECCIÓN DEL CURSO:
# PROFESOR DE TEORÍA:
# PROFESOR DE LABORATORIO:
#
# AUTOR
# NOMBRE:
# RUN:
# CARRERA:
```

Suba su solución al apartado especificado en UVirtual en la pestaña «EVALUACIONES», apartado «[E1] — PREGUNTA 1» o «[E1] — PREGUNTA 2», según corresponda.

1. Juan S.A.

Juan posee una empresa que debe realizar pedidos a todo el mundo. En un archivo llamado `data.csv` está la información de todos sus proveedores, la que incluye:

id El ID del proveedor, correspondiente a tres (3) letras que identifican al país, seguido de dos (2) series de números, cada campo separado por guiones. Por ejemplo, un identificador para una empresa chilena tendría la forma: CHL-1974-714030.

nombre Nombre del proveedor. Todo proveedor necesita un nombre.

apellido Apellido del proveedor. Complementa su nombre.

activo Señala si el proveedor sigue activo o no.

company Nombre de la compañía del proveedor.

mail E-mail del proveedor.

latitud Latitud de la ubicación del proveedor (ángulo entre el plano ecuatorial y la ubicación en sí).

longitud Longitud de la ubicación del proveedor (ángulo entre el plano del meridiano de referencia y la ubicación en sí).

fecha_entrega Fecha en la que se debía entregar el pedido, en formato año/mes/día, por ejemplo, 2024/5/25.

ultima_entrega Fecha de la ultima entrega, también en formato año/mes/día.

cantidad Cantidad del producto solicitado en toneladas.

monto Valor del pedido.

Juan necesita conocer qué proveedores siguen activos en un determinado país y si es que tienen algún pedido pendiente, para recordarles este pedido y que deben entregarlo. Para ello, Juan te pide que generes un programa al que él ingresará el código del país y el programa debe determinar cuáles proveedores activos tienen un atraso en su entrega, para mostrarle a Juan qué correos debe enviar.

Para determinar si un proveedor está atrasado, se debe comparar la fecha de la última entrega y ver si esta es previa a la fecha del pedido. De ser así, significa que el proveedor está atrasado con el pedido.

1.1. Entrada

La entrada la pedirá el programa con el mensaje «Ingrese el codigo del pais: » y esta corresponderá al código de país (tres letras en mayúsculas):

Ingrese el codigo del pais: <Código>

Por ejemplo,

Ingrese el codigo del pais: CHL

1.2. Salida

La salida corresponde a los correos que debe mandar Juan, los que tienen el siguiente formato:

```
from: the_juan@company.cl
to: <Mail del proveedor>
subject: Atrazo!
body: Apurenses ph!
Besitos cordiales,
El Juan.
```

Estos correos deben aparecer por pantalla, separados entre sí por una línea en blanco (pero sin líneas en blanco antes o después de estos).



1.3. Ejemplo

Para la entrada:

Ingresa el código del país: CHL

La salida completa utilizando el archivo `data.csv` es:

```
from: the_juan@company.cl
to: maighdiln.hommell@bluepeak.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.
```

```
from: the_juan@company.cl
to: hillary.wurst@momentumtech.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.
```

```
from: the_juan@company.cl
to: marjorie.eckstein@brightbridge.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.
```

```
from: the_juan@company.cl
to: antonia.mcclusky@primeedge_solutions.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.
```

```
from: the_juan@company.cl
to: delores.furrow@digitalpath.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.
```

```
from: the_juan@company.cl
to: inge.boey@purefusion_labs.com
subject: Atrazo!
body: Apureense ph!
```



Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: gale.knutson@bluerise.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: madelina.wohlert@novatech.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: lita.verkler@swiftvision_labs.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: alys.kellar@futuruslabs.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: inez.albornoz@swiftcore.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

from: the_juan@company.cl
to: estel.griffitt@greenwave.com
subject: Atrazo!
body: Apureense ph!
Besitos cordiales,
El Juan.

2. Mortar Combat

La compañía de mercenarios *Band of Dropouts* está analizando sus tácticas y para ello, pretenden utilizar un software que construirán a partir de varias partes. En particular, necesitan que realices una función que, a partir de datos que representen un mapa con la posición de tanques y los disparos hechos sobre este, determine los aciertos en el ataque.

La función que debes desarrollar se llamará **contar** y tomará dos parámetros:

- Una lista de listas de *strings*, la que representará con "_" espacios vacíos en el mapa y con "*", la silueta de los tanques. Nota que no es un tanque por posición, sino que varias posiciones podrían representar la silueta de uno.
- Una lista de *strings* que tienen los lugares del mapa que han sido atacados, correspondientes a coordenadas separadas por coma, en formato "<fila>,<columna>", por ejemplo, "5,6".

Dado que la unidad de morteros de la compañía, encargada de realizar estos ataques, estaba llena de reclutas nuevos y nerviosos, no todos los ataques cayeron dentro de los confines del mapa, por lo que tienes que considerar esto al momento de programar tu solución.

2.1. Prototipo de la Función y Entrada

La función, en definitiva, debe poder ser llamada con el siguiente formato:

```
contar(<mapa>, <ataques>)
```

donde <mapa> corresponde a la lista de listas que representa el mapa y <ataques>, la lista con las coordenadas de los ataques realizados por la unidad de morteros.

2.2. Salida

La salida de la función corresponde a un entero, la cantidad de aciertos que tuvieron sobre tanques los morteros con sus ataques, es decir, debe retornar un entero.

2.3. Consideraciones

- Puedes crear un bloque principal para probar tu función con entradas que conozcas o plantees tú mismo o misma, pero la entrega no puede tener nada más que la función solicitada, es decir, sin un bloque principal ni código adicional (salvo funciones auxiliares que puedas necesitar). No necesitas borrarlo en la entrega, pero sí, al menos, comentarlo.
- Recuerda respetar el orden de los parámetros formales, pues estos son posicionales.

2.4. Ejemplo

Para las entradas:

```
mapa = [
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '*', '_', '_', '_', '*', '_', '_', '*', '_', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_', '*', '*', '*', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_', '*', '*', '*', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '*', '_', '_', '_', '*', '_', '_', '_'],
    ['_', '_', '*', '*', '*', '_', '*', '*', '*', '_', '_'],
    ['_', '_', '*', '*', '*', '_', '*', '*', '*', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_', '_']
]

ataques = ['2,14', '12,2', '9,11', '6,2', '9,2', '6,3', '3,8',
            '13,1', '16,0', '13,14', '10,6', '13,4', '13,3', '12,5',
            '15,6', '16,5', '9,4', '6,14', '0,4', '3,2', '7,6', '2,12',
            '7,1', '15,9', '15,7', '9,3', '10,8', '1,8', '7,5', '15,1',
            '4,3', '3,13', '13,11', '13,7']
```

el llamado:

```
contar(mapa, ataques)
```

debe retornar 4.

Para las entradas:

```
mapa = [
    ['_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '*', '_', '_', '_', '*', '_', '_', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '*', '_', '_', '_', '*', '_', '_', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '*', '_', '_', '_', '*', '_', '_', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_'],
    ['_', '_', '_', '_', '_', '_', '_', '_', '_'],
    ['_', '*', '_', '_', '_', '*', '_', '_', '_'],
    ['*', '*', '*', '_', '*', '*', '*', '_']
]
```



```
    ['*', '*', '*', '-', '*', '*', '*', '-', '-', '-'],
    ['-', '-', '-', '-', '-', '-', '-', '-', '-', '-'],
    ['-', '-', '-', '-', '-', '-', '-', '-', '-', '-'],
    ['-', '-', '-', '-', '-', '-', '-', '-', '-', '-']
]
ataques = ['7,11', '6,5', '21,6', '22,11', '6,10', '6,7',
            '0,6', '4,5', '3,8', '22,7', '7,12', '12,3', '19,7',
            '17,6', '21,8', '13,6', '14,3', '11,7', '21,9', '13,4',
            '15,1', '1,5', '12,0', '12,5']
```

el llamado:

```
contar(mapa, ataques)
```

debe retornar 5.