



UNIVERSIDAD ARGENTINA DE LA EMPRESA
Departamento de Tecnología
Informática

Introducción a la algoritmia (CODIGO)

Profesor: Anabella Baer

Examen Final

ALUMNO:

LU:

FECHA: 06/12/2024

CARRERA:

TEMA: 1

NOTA: EL EXAMEN ESCRITO ES UN DOCUMENTO DE GRAN IMPORTANCIA PARA LA EVALUACIÓN DE LOS CONOCIMIENTOS ADQUIRIDOS, POR LO TANTO, SE SOLICITA LEER ATENTAMENTE LO SIGUIENTE:

- Responda claramente cada punto, detallando con la mayor precisión posible lo solicitado.
- Aprobación del examen: Con nota mayor o igual a 4 (cuatro)
- Duración de examen: 2 horas
- **Se deberá cargar el o los archivos ".py" a la tarea de Teams creada. Tiene horario de finalización, por lo tanto, no se pueden recibir archivos de otra forma. Por favor tener en cuenta el tiempo ya que el examen puede quedar sin entregar.**
- Condiciones de aprobación y evaluación:
 - 1) Interpretación del enunciado. En caso de asumir definiciones del contexto del problema, aclararlas en forma explícita en el examen (a modo de comentario).
 - 2) Desarrollo de la estrategia y lógica que resuelva el o los problemas planteados y el código Python que lo representa.
 - 3) Respeto de buenas prácticas de programación vistas en clase.
 - 4) Se evaluará la calidad de la interfaz de interacción con el usuario, es decir, como es la comunicación entre el algoritmo y el usuario con el que interactúa.
 - 5) No se aceptarán ejercicios que incluyan funciones o herramientas de Python no tratados en clase.

NOTA EN NÚMEROS

NOTA EN LETRAS

SELLO

FIRMA DEL DOCENTE

Ejercicio:

Una escuela secundaria necesita un programa para administrar las calificaciones de los alumnos de un curso.

El programa debe realizar las siguientes tareas:

1. Leer por teclado las notas y el legajo (4 dígitos, no repetidos) de los alumnos. Finaliza la lectura de los datos, cuando se lee un legajo igual a -1.
2. Determinar cuántos alumnos totales hay. Imprimir por pantalla.
3. Determinar cuántos alumnos reprobaron, nota menor que 4. Imprimir por pantalla los legajos de cada uno y el número total.
4. Imprimir la lista total de las notas y los legajos tal como fue cargada y luego volver a imprimirlas ordenadas de menor a mayor por número de legajo.
5. Utilizar al menos una función (además de la de ordenamiento) para resolver el enunciado.
6. El programa debe realizar todas las validaciones correspondientes a la lectura y procesamiento de datos.

Se dispone de la función de ordenamiento genérica para utilizar en el algoritmo de resolución del enunciado. En caso de ser necesario, adaptarla a las necesidades del problema.

#Funcion de ordenamiento genérico por burbujeo

```
def metododeintercambio(lista):  
    desordenada = True  
    while desordenada:  
        desordenada = False  
        for i in range(len(lista)-1):  
            if lista[i]>lista[i+1]:  
                aux = lista[i]  
                lista[i] = lista[i+1]  
                lista[i+1] = aux  
                desordenada = True
```