

RÚBRICA QUE EVALÚA AL INSTRUMENTO: PROYECTO DE FIN DE SEMESTRE, EDA I

Representante de equipo: (Para llenar por el profesor)		Atributos de egreso CACEI: A4-CD1			
Semestre:		Resultados de aprendizaje ANECA: 1.2, 2.2			
Atributo de evaluación	Puntos	10 (Excelente)	8 (Bien)	6 (Suficiente)	0 (Insuficiente)
ANÁLISIS	10	Diseño e implementó 2 tipos abstractos de nivel superior.	Diseño e implementó 1 tipo abstracto de nivel superior.	Diseño e implementó 2 tipos compuestos de nivel superior.	Diseño e implementó 1 o ningún tipo compuesto de nivel superior.
	20	Las APIs de sus tipos abstractos de nivel superior están completas al 100%. (Completa significa que el cliente no tendrá que combinar operaciones porque alguna de uso común no fue implementada.)	Las APIs de sus tipos abstractos de nivel superior están completas al 80%. (El cliente tiene que combinar operaciones porque alguna de uso común no fue implementada.)	Las APIs de sus tipos abstractos de nivel superior están completas al 60%.	Las APIs de sus tipos abstractos de nivel superior están completas en menos del 60%.
	20	Utilizó tipos abstractos para todas las otras entidades que forman a su aplicación.	Utilizó tipos abstractos para casi todas las otras entidades que forman a su aplicación.	Utilizó tipos abstractos para algunas de las otras entidades que forman a su aplicación.	Utilizó tipos compuestos para una o más de las otras entidades que forman a su aplicación.
	10	Utilizó al menos uno de los siguientes algoritmos: PD, BT.	Utilizó al menos uno de los siguientes algoritmos: Dividir y vencer, Voraz.	Utilizó fuerza bruta.	No utilizó ningún algoritmo.
IMPLEMENTACIÓN	15	Utilizó listas doblemente enlazadas para implementar listas, pilas y colas.	Utilizó arreglos dinámicos para implementar listas, o pilas, o colas.	Utilizó arreglos estáticos para implementar listas.	Utilizó arreglos estáticos para implementar listas.
	10	Todas y cada una de las estructuras de datos que utilizó están en su propio módulo.	La mayoría de las estructuras de datos que utilizó están en su propio módulo.	Algunas de las estructuras de datos que utilizó están en su propio módulo.	Dos o más de las estructuras de datos que utilizó NO están en su propio módulo.
	10	Las APIs de sus tipos abstractos de nivel superior están documentadas al 100% con Doxygen.	Las APIs de sus tipos abstractos de nivel superior están documentadas al 90% con Doxygen.	Las APIs de sus tipos abstractos de nivel superior están documentadas en menos del 80% con Doxygen.	Las APIs de sus tipos abstractos de nivel superior están documentadas en menos del 80% con Doxygen, o utilizó un formato diferente a éste.
	5	Su código fuente está en BitBucket, GitHub, o similar. (Incluye el archivo instrucciones.txt donde está la instrucción de compilación.)	Su código fuente está correctamente empacado en un archivo comprimido. (Incluye el archivo instrucciones.txt donde está la instrucción de compilación.)		Envió archivos sueltos, o NO pudo encontrar el archivo con las instrucciones de compilación.
Generales					
EL PROYECTO NO COMPILA	-10	Si el proyecto no compila en Linux, por la razón que sea, no lo tomaré en cuenta.			
FUGAS DE MEMORIA	-1	De la calificación obtenida quitaré 1 punto por cada fuga de memoria encontrada.			
PROBLEMAS CON LA MEMORIA	-5	De la calificación obtenida quitaré 5 puntos si el programa se detiene por un problema de memoria.			
REPORTE	-2	De la calificación obtenida quitaré 2 puntos si el reporte no está debidamente llenado.			
COMPILACIÓN	-5	De la calificación obtenida quitaré 5 puntos si no encuentro la instrucción de compilación.			