

Nom:	<u>Ús</u>
Curs:	

Temps estimat	Tipus d'activitat
---------------	-------------------

2,5 hores

Avaluació

Normes

- El codi lliurat ha de compilar sense problemes
- De cada exercici tindrà la pròpia nota, s'especificarà la nota mínima. Si aquesta nota mínima no s'assoleix, la nota de l'examen serà 4 com a màxim.
- De cada exercici s'ha de proporcionar el main que has fet servir per veure el funcionament correcte. Si aquest main és escàs i es deixa porcions d'algorisme sense cobrir baixarà la nota. Si és massa escassa, la nota màxima serà de 4.
- L'estructura de cada fitxer serà el següent (si no, no es corregirà l'exercici)
 - enums? delegats?
 - Una classe
 - Classes noves dins de la classe
 - Atributs
 - Propietats
 - Constructors
 - Destructors
 - Getters i Setters
 - Altres mètodes
- Si cometeu algun dels errors comentats a classe (que són majoritàriament 3) que no es poden permetre, la nota màxima de l'examen serà de 4.
- No pots fer trampes



(1 punt) (Nota mínima 7) Implementa la funció següent: Implementa una funció que, donat un array de sencers, et retorne, la posició del valor mínim, la posició del valor màxim i la mitjana de l'array. La mitjana d'un array d'enters és el valor que es troba al centre de l'array quan els seus elements estan ordenats de menys a més gran; si l'array té un nombre parell d'elements, la mitjana és la mitjana dels dos valors centrals. Pots fer trampes, ..., una mica

(2 punts) (Nota mínima 7) Implementa la funció següent.

Escriu una funció que rebrà una llista de nombres enters i torneu una llista de parells de nombres que sumen un nombre objectiu donat. Per exemple:

nombres = [2, 4, 3, 5, 7, 8, 9]

objectiu = 10

Resultat: [(2, 8), (3, 7)]

(2 punts) (Nota mínima 4) Implementa la classe següent: Implementa una classe anomenada Cap de tasques que gestione una llista de tasques. Cada tasca es representarà amb una instància de la classe Task, i són immutables, és a dir, un cop creada una tasca no es pot canviar.

Tasca de la classe

- Id, nom, descripció, data de realització, prioritat i estat de la tasca (que no pot ser un string)
- Pel que fa a les dates
 - DateTime dateTime1 = dataHora nova (2023, 6, 15);
 - Es poden fer servir els operadors normals

TaskManager de classe

- **Propietats**
 - TaskCount
 - MaxDate: Tornarà la data major
- **Constructors**
 - Un únic constructor amb el nombre màxim de tasques que podeu emmagatzemar.
- **Mètodes**
 - Els que necessitis
 - Afegir tasca
 - ContainsTask(string nom de la tasca)
 - FilterTask(delegat): Retornarà una llista ordenada amb totes les tasques que passen el filtre. Podeu utilitzar les funcions de sort de c#
 - ClearTasks
 - RemoveAllTasks(string): El string d'aquesta funció serà el nom de la tasca o el valor de la descripció
 - Si a alguna de les funcions se li passa una cosa que no és vàlida, haureu de llançar una excepció.

(2 punts) (nota mínima 3) Implementa la classe següent, posa els atributs, constructors i mètodes que vegis necessaris. La classe que cal implementar és un diccionari genèric. Entre tots els mètodes i atributs que ha de tenir un diccionari per funcionar, ha de tenir 3 funcions: una que torni totes les keys, una altra que torni tots els valors i una altra que visiti totes les entrades amb un delegat.