



Universidad del Istmo de Guatemala
Facultad de Ingenieria
Ing. en Sistemas
Informatica II
Prof. Ernesto Rodriguez - erodriguez@unis.edu.gt

Laboratorio #4

Fecha de entrega: 27 de Febrero, 2019 - 11:59pm

Instrucciones: Resolver cada uno de los ejercicios siguiendo sus respectivas instrucciones. El trabajo debe ser entregado a traves de Github, en su repositorio del curso, colocado en una carpeta llamada "Laboratorio #3". Este laboratorio debe ser elaborado en parejas.

Tarea #1 (25%)

Defina la función **bool** *divisionSegura*(**int** *numerador*, **int** *denominador*, **int*** *respuesta*). Esta función debe calcular la division entre dos numeros validando que no se este utilizando 0 en el denominador. En caso que la division sea posible, esta función debe hacer la division y guardarla en el puntero "respuesta" y retornar **true** de lo contrario, debe retornar **false**.

Tarea #2 (25%)

Defina una función llamada "**bool** *sumaMayor*(**int**[] *valores*, **int** *cantidad*, **int**** *respuesta*)". Esta función debe aceptar un arreglo de numeros enteros y buscar los dos numeros que generen la mayor suma en este arreglo. Estos dos numeros se deben almacenar en el puntero que se recibe como segundo parametro utilizando el indice 0 para el primer valor y el indice 1 para el segundo valor. Esta función retorna **true** cuando el arreglo recibido tiene al menos dos elementos, de lo contrario, no se hace ninguna operación y se retorna false.

Tarea #3 (40%)

Defina la función **void** *fibonacciN*(**constint** *n*, **int*** *valores*). Esta función acepta un numero "n", produce los primeros *n* numeros de fibonacci y los almacena en el arreglo "valores" que se le ha dado como parametro.

Tarea #4 (10%)

Optimize la función "*fibonacciN*" de la tarea 3 de tal forma que pueda calcular los numeros de fibonacci eficientemente. Puede probar que su implementación funciona utilizando un numero grande como parametro. Por ejemplo, calcular los primeros 10,000 numeros de fibonacci.