*RESUMEN PREGUNTAS VERDADERO O FALSO PARA FINAL DE CADP*

toda solución correcta es eficiente?

*Falso, siempre dependerá de la cantidad de memoria consumida*

------------------------------------------------------------------------------------------------------------------

las estructuras de datos elegidas determinar que una solución sea eficiente?

Para mí es Falso, la correcta elección de la estructura de datos puede llevar a una solución eficiente, pero el uso que le des es realmente lo que lleva a que sea eficiente. Por ejemplo, si tenemos 2 medios de pago, los guardamos en una lista y tuvieramos que obtener todas las ventas de un mismo medio de pago, podemos usar la lista de forma ordenada para que consuma menos tiempo la búsqueda. Pero la eficiencia de esto no depende tanto de la lista sinó de su uso

siempre se puede transformar un function en un módulo procedure?

*Verdadero, el valor a retornar puede pasarse como un parámetro de referencia*

--------------------------------------------------------------------------------------------------------------------

un programa correcto asegura la eficiencia?

*Falso, el programa puede ser correcto pero tal vez no eficiente*

---------------------------------------------------------------------------------------------------------------------

un programa bien documentado asegura eficiencia?

Falso, un programa bien documentado asegura una mejor legibilidad, pero la eficiencia no depende de la documentación sinó del uso de memoria y tiempo"

-------------------------------------------------------------------------------------------------------------

un módulo función no puede retornar un tipo de dato puntero a un arreglo

------------------------------------------------------------------------------------------------------------

si p es una variable de tipo puntero, entonces no es válida la siguiente instruccion readln(p^)

Falso, que p sea una variable de tipo puntero no condiciona la validez de la instrucción. Lo que lo condiciona es el tipo de dato al que el puntero apunta. Si p apuntara, por ejemplo, a un real o a un entero, sería correcto. Pero si apuntara a un registro sería incorrecto

--------------------------------------------------------------------------------------------------------------

si un programa utiliza variables globales entonces no puede contener modulos que utilicen parámetros

falso, se pueden utilizar modulos

-------------------------------------------------------------------------------------------------------------

un modulo con 5 lineas de codigo es mas eficiente en tiempo de ejercucion que un programa con 100 lineas de código

Falso, depende el problema que ambos códigos quieran solucionar. Por ejemplo, las 100 lineas de código podrían terminar más rápido que las 5 lineas de código en el caso que las 5 lineas fueran un for de 1 hasta 1 millón y las 100 lineas de código sean solo "writeln("Hola")"

--------------------------------------------------------------------------------------------------------------

un tipo de dato registro puede ser homogeneo si todos sus campos son del mismo tipo de dato

Es Falso porque la heterogeneidad viene de si la estructura puede tener o no datos de distintos tipo. Como los registros sí pueden tener datos de distinto tipo, es heterogenea

--------------------------------------------------------------------------------------------------------------

nunca es posible acceder al nodo de la posicion n en una estructura del tipo lista

Falso, las listas son estructuras de datos de acceso secuencial, lo que requiere que para llegar a la pos n, se acceda primero a todos los nodos que anteceden esa posición.

------------------------------------------------------------------------------------------------------------

incluir modulos dentro de un programa implica que el programa es mas eficiente que otro programa que realiza la misma tarea pero sin utilizar modulos

----------------------------------------------------------------------------------------------------------------------

no siempre es posible declarar un tipo subrango donde su tipo base sea cualquiera de los tipos simples vistos en la teoría

-----------------------------------------------------------------------------------------------------------------------

un programa que utiliza repeat-until puede reescribirse utilizando un while

----------------------------------------------------------------------------------------------------------------------

la comunicacion entre el programa y los modulos no solo se puede hacer utilizando parámetros

-----------------------------------------------------------------------------------------------------------------------

siempre es posible realizar la eliminacion de un elemento en el vector

*falso, el elemento a eliminar puede no existir*

--------------------------------------------------------------------------------------------------------------------

el acceso a un elemento de una estructura de datos lineal solo es posible a traves de un recorrido secuencial

-------------------------------------------------------------------------------------------------------------------------

No es posible la utilización de las variables globales para la comunicación entre los módulos de un programa.

*Falso, es posible pero no eficiente*

----------------------------------------------------------------------------------------------------------------------

un programa que contiene errores semanticos se puede compilar

*verdadero, puede que en la compilación no haya errores pero al momento de ejecutarlo si*

----------------------------------------------------------------------------------------------------------------------

un vector siempre es almacenado en memoria estatica

----------------------------------------------------------------------------------------------------------------------

una variable de tipo puntero siempre se almacena en memoria dinámica

----------------------------------------------------------------------------------------------------------------------

las listas se clasifican como estructuras heterogéneas

*falso, las listas son estructuras homgeneas*

-----------------------------------------------------------------------------------------------------------------------

se puede agregar un elemento del final de una lista sin necesidad de realizar un recorrido secuencial de la misma

----------------------------------------------------------------------------------------------------

el tipo base de un tipo de dato subrango puede ser un tipo de dato real

*falso, debe ser ordinal*

--------------------------------------------------------------------------------------------------------

la seccion type en un programa únicamente se puede incluir luego de la declaracion de las constantes y antes de la declaración de los modulos

Falso, dentro de los módulos también se puede, ya que son una unidad autocontenida

----------------------------------------------------------------------------------------------------------

un programa correcto con 3 módulos es más eficiente que un programa correcto con 5 módulos

Falso, la eficiencia dependerá del tiempo y memoria que consuman. Además pueden ser dos programas distintos que resuelvan problemas distintos así que la cantidad de módulos no condiciona la eficiencia.

------------------------------------------------------------------------------------------------------------

en la tecnica de corrección de debuggin es necesario analizar los casos limites del problema

*falso, los casos limites solo se analizan la técnica de correccion llamada testing*

*-----------------------------------------------------------------------------------------------------------*

una función puede devolver un tipo de dato registro, real, booleano, integer, entre otros

*falso, solo puede devolver una tipo de dato simple, registro no lo es.*

*------------------------------------------------------------------------------------------------------------*

antes de utilizar una variable puntero siempre se debe reservar memoria

-------------------------------------------------------------------------------------------------------------

las instrucciones dentro de una estructura de control repeat-until se pueden ejecutar 0,1, o más veces

*Falso, se ejecuta 1 o mas veces*

---------------------------------------------------------------------------------------------------------------

las instrucciones dentro de una estructura de control for siempre se ejecutan al menos 1 vez

----------------------------------------------------------------------------------------------------------------

en el acceso a los campos de un registro es necesario respetar el orden que fueron declarados

*falso, su acceso es directo, independientemente de su orden*

*----------------------------------------------------------------------------------------------------------------*

una variable global solo puede ser accedida y modificada desde el cuerpo del programa principal

*falso, puede ser modificada desde cualquier modulo ya que es una variable global*

-----------------------------------------------------------------------------------------------------------------

para utilizar una variable del tipo puntero p siempre se debe realizar new(p)

-------------------------------------------------------------------------------------------------------------------

se pueden utilizar operaciones de entrada/salida sobre todos los campos de una variable de tipo registro

---------------------------------------------------------------------------------------------------------------------

al asignar el valor de nil a un puntero se libera la memoria referenciada

*falso, deja al puntero sin dirección asignada pero el espacio de memoria queda ocupada*

*------------------------------------------------------------------------------------------------------------------*

todos los modulos de tipo funcion de un programa se pueden escribir como modulos de tipo procedimiento

VERDADERO, porque la funcion retorna un valor y un procedimiento tambien puede, lo q no se puede es si es al reves xq el proceso puede no devolver nada mientras q la funcion si o si