Portafolio

Algoritmia

Facultad de matemáticas

Antonio Alfonso Cetzal Patrón. LCC.

# Reflexión

La importancia de esta asignatura en mi carrera es bastante, ya que para nosotros es la primera interacción que tenemos programando, eso si algunos con anticipación no habían aprendido a programar, la asignatura es la base de nuestra formación, en esta aprendemos lo que yo podría decir que es lo básico, esto es de los bucles, los tipos de variables, las funciones que podemos crear, en mi caso, esta es la segunda vez que tomo esta materia, la primera vez que la tome falle en no poder hacer bastantes practicas para poder acostumbrarme, principalmente en estas dos ultimas unidades que fueron los arreglos unidimensionales y bidimensionales, lo importante en esta asignatura es llevar a los alumnos que por lo menos unos cuantos semestres tengan el ámbito de poder analizar tranquilamente un problema y así poder llevarlo a la solución.

Para mí, el tema que mas me gustó fue el de arreglos unidimensionales, el porque es que para mi fue un reto plasmar mis ideas que tenía a un algoritmo, ya estaba bastante familiarizado con el lenguaje C o C++ que al momento de volver a ver algoritmia para mi fue un problema, ya que no todo lo que se puede hacer en otros lenguajes de programación se puede expresar directamente a un algoritmo, también en el programa que utilizábamos hay bastantes expresiones que podemos usar para facilitar el proceso, de este tema las actividades que mas me gustaron fue el de la criba, ya que este problema nunca lo había viste y me llamo bastante la atención, otro que igual me gusto fue el de encontrar si existe una subcadena en una cadena, es otro problema que hasta en ese momento no se me había planteado, este me dio bastantes problemas ya que no encontraba la solución.

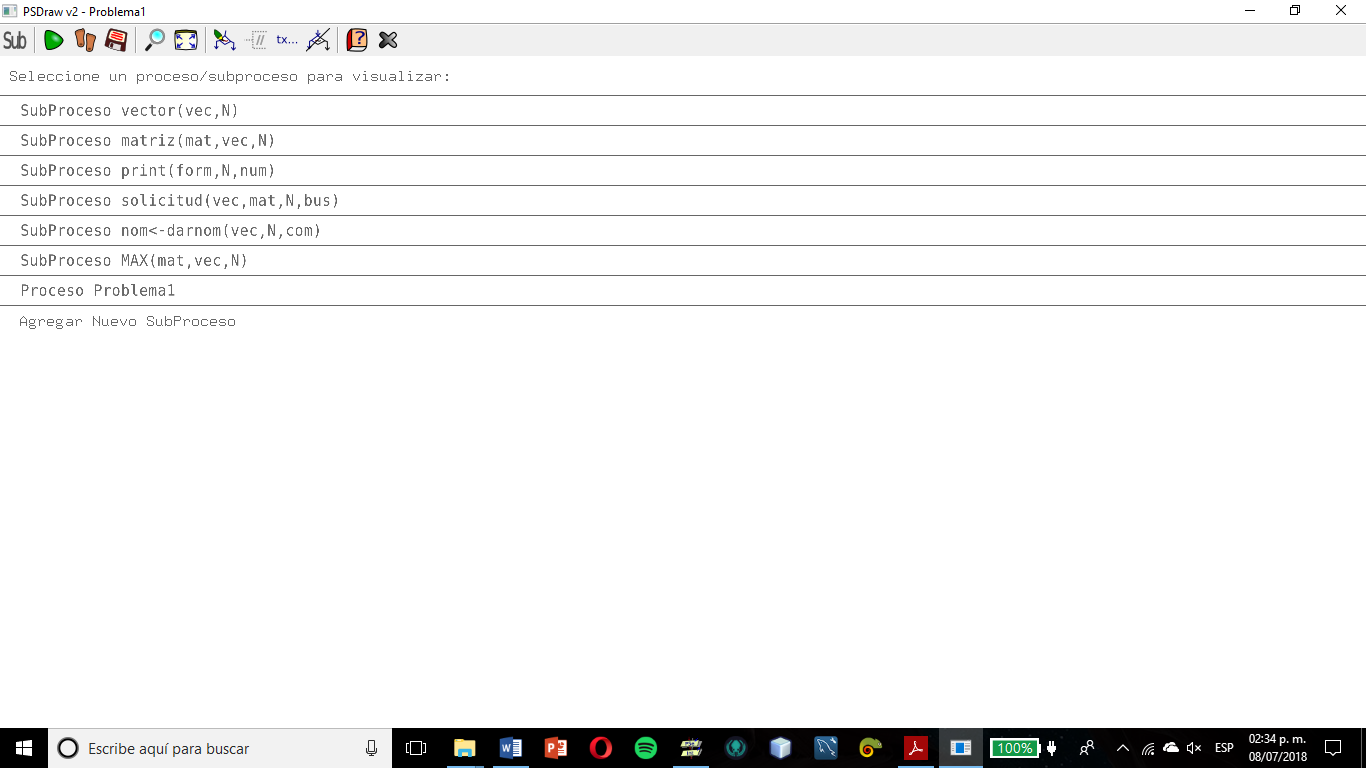
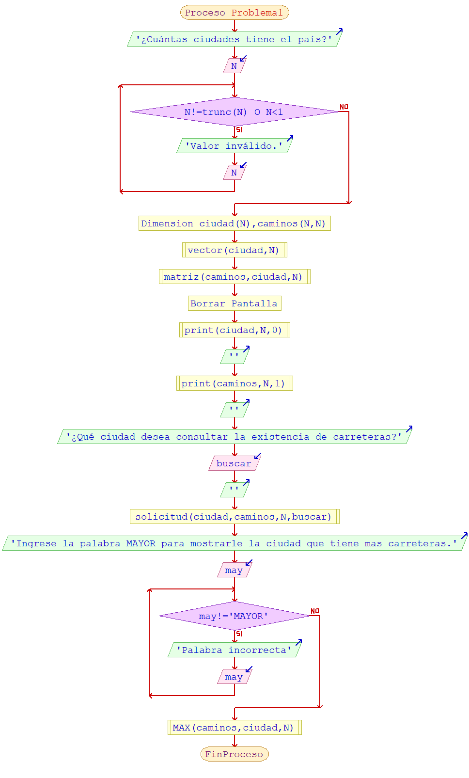
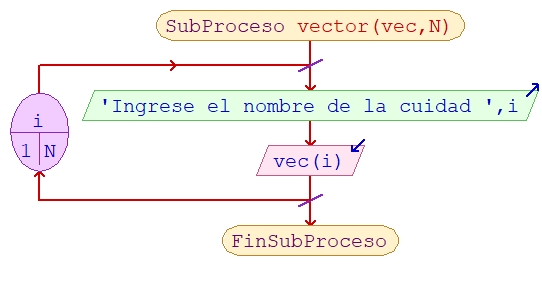
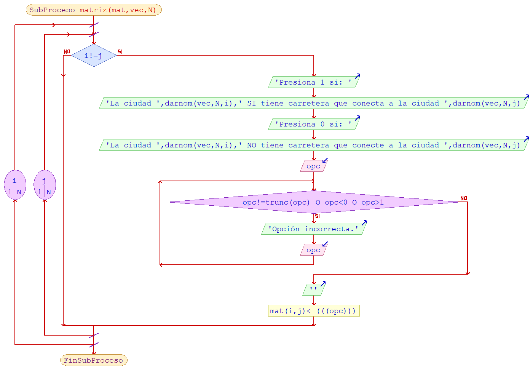
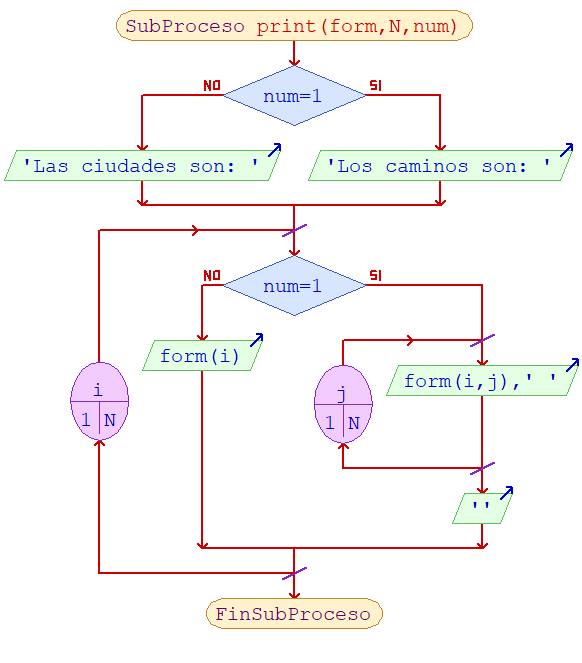
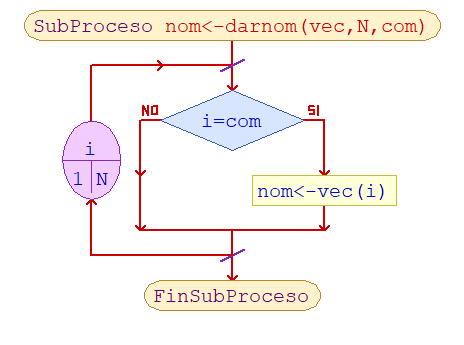
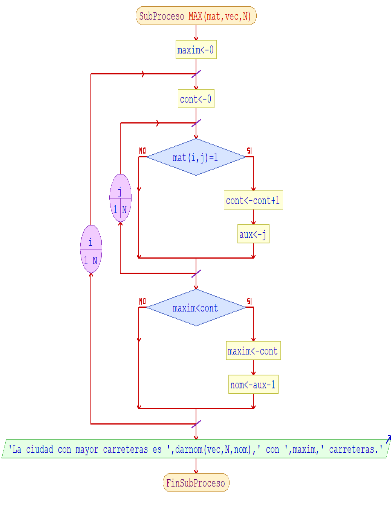
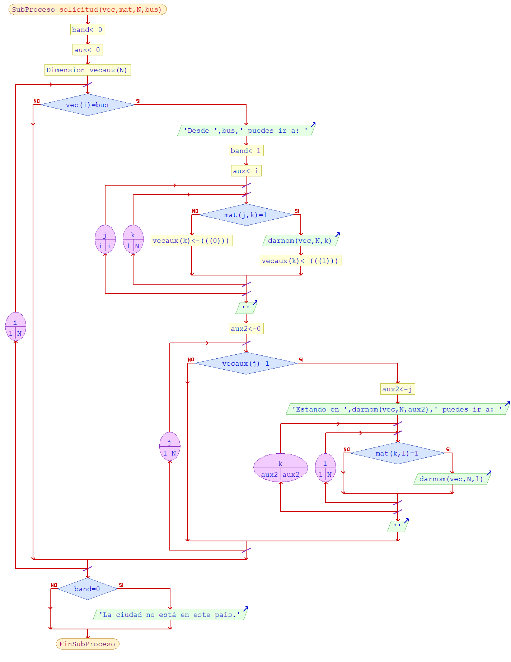
Los temas que menos me gustaron fueron los 2 principales, ya que eran bastante básicos, solo por eso considero que fueron los menos importantes para mí, pero las actividades que nos ponía hacer el maestro de esa unidad fueron bastantes intuitivas y divertidas.

Lo que me ha dejado este curso como fortaleza principal es poder sentarme y analizar tranquilamente el problema y así poder ahorrarme tiempo que saltarme eso e ir directo a la programación, en algunos casos lo sigo aplicando ya que se cómo hacer el problema.

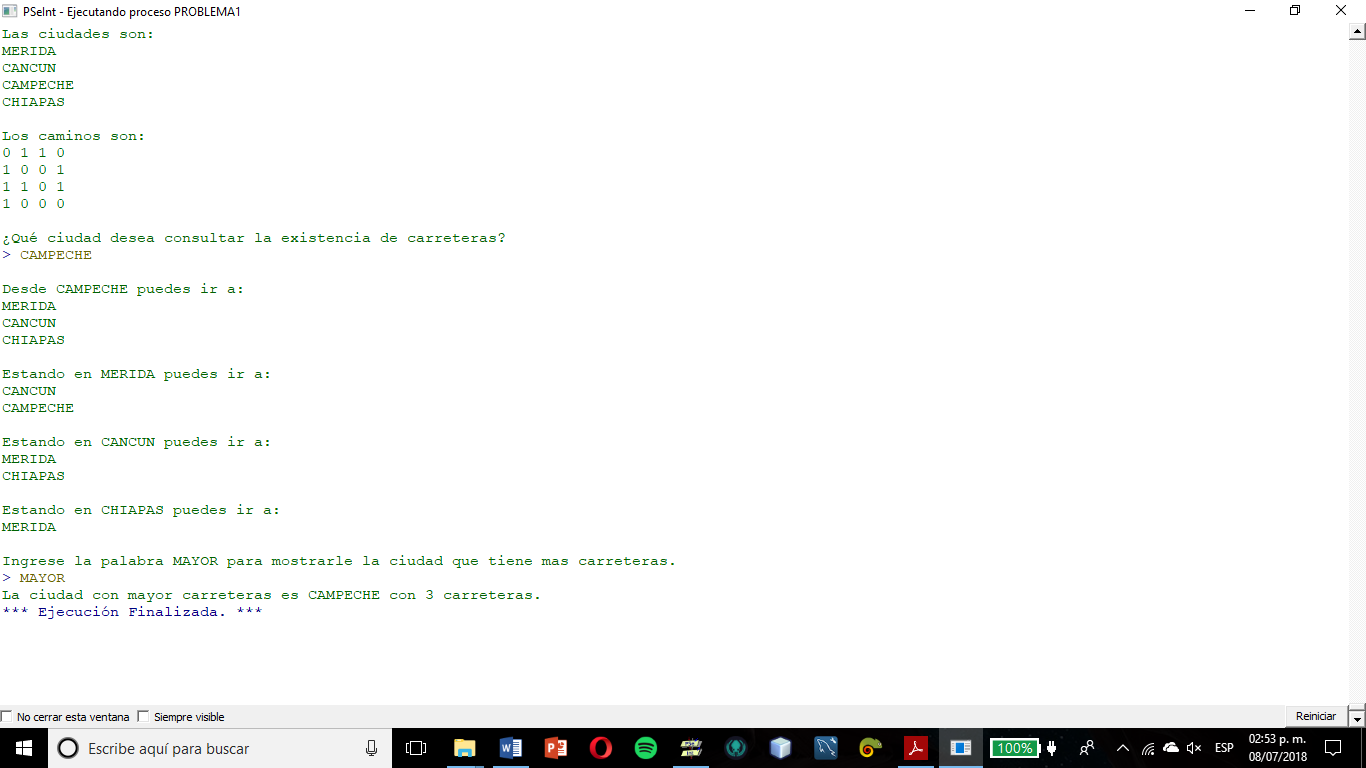
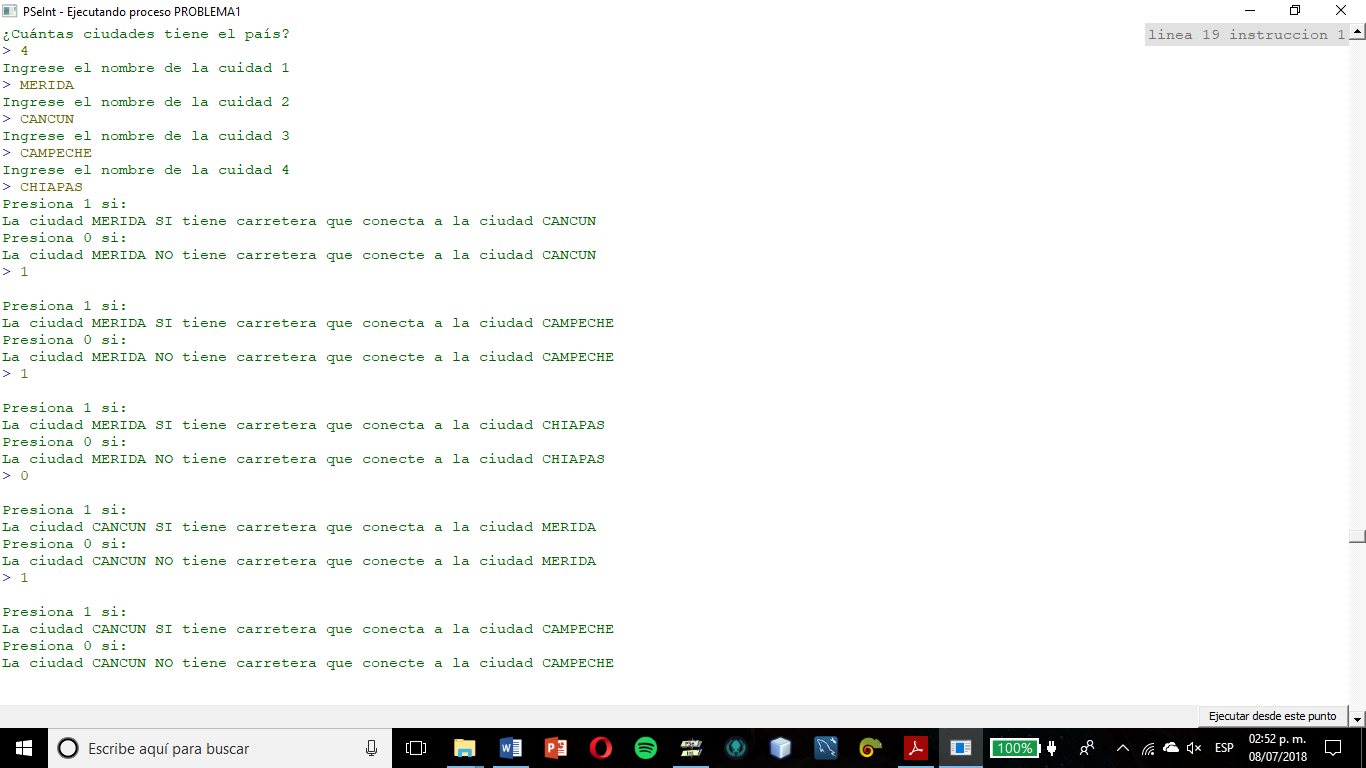
La debilidad que puedo destacar mas es la forma de pensar el algoritmo, ya que como dije anterior, estoy bastante familiarizado con otros lenguajes de programación, y pasar mis ideas que tengo de como programarlo en esos lenguajes a algoritmos se me complica bastante.

Cada maestro utilizo distintas formas de enseñarnos sus temas, el primer maestros dado los temas que le toco explicar pues fueron mas actividades que nos hizo hacer, las cuales fueron bastante interactivas, fáciles y también divertidas, el segundo profesor usaba más una estrategia combinada, ya que si nos marcaba ejercicios con bastante nivel pero también nos explicaba como hacerlos, nos proyectaba el pseudocódigo o diagramas de flujo, el ultimo profesor uso una estrategia que podría decir que fue, mas analista, es decir, que nos hacia que pensemos los problemas antes de entrar a la codificación, la forma de evaluar sentí que fueron correctas, aunque usaron distintas formas, el primer profesor nos evaluó por medio tareas y un pequeño examen con 3 problemas, el segundo nos califico de una forma presencial, ya que cada día había tarea nueva y teníamos que hacerla y entregarla para poder obtener nuestras calificaciones, el tercero uso lo mismo que el primero y adicionalmente anexo el portafolio, supongo que el anexar el portafolio esta bien ya que es el último tema necesitaría evaluar todo lo aprendido, ya que en estas dos últimos temas se aplicaba todo lo que habíamos visto antes del curso.

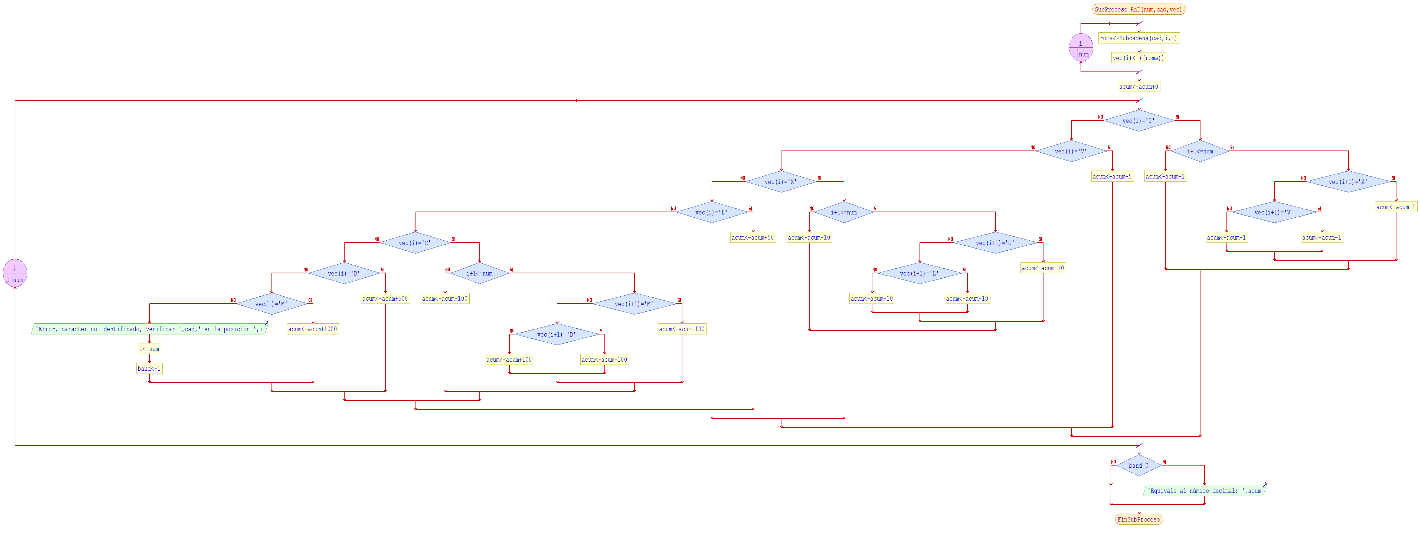
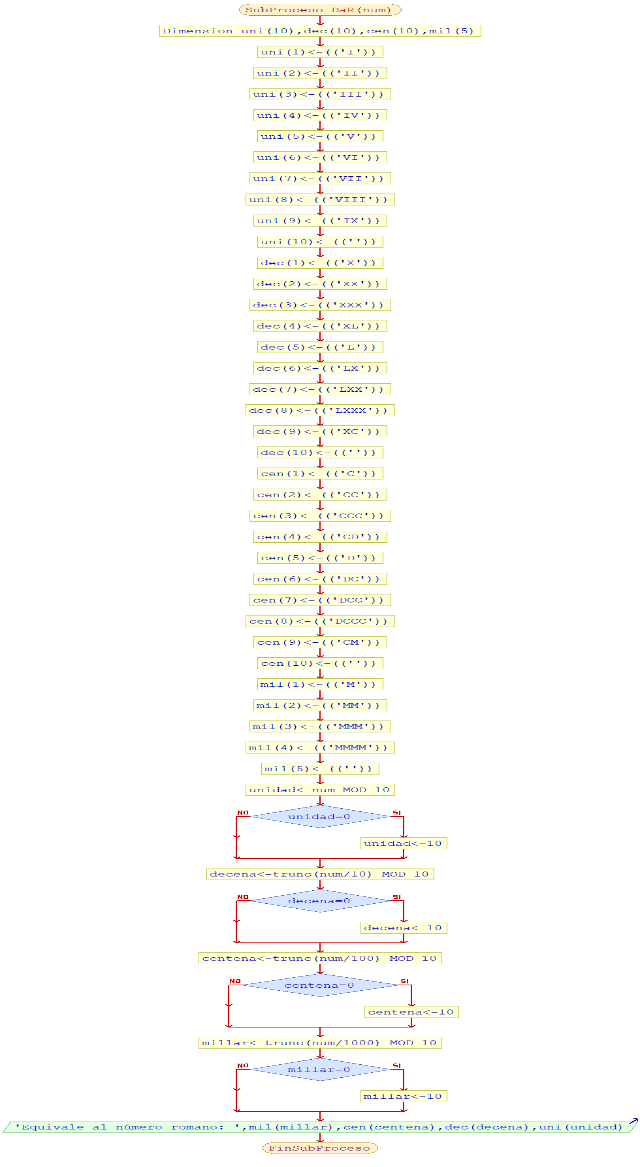
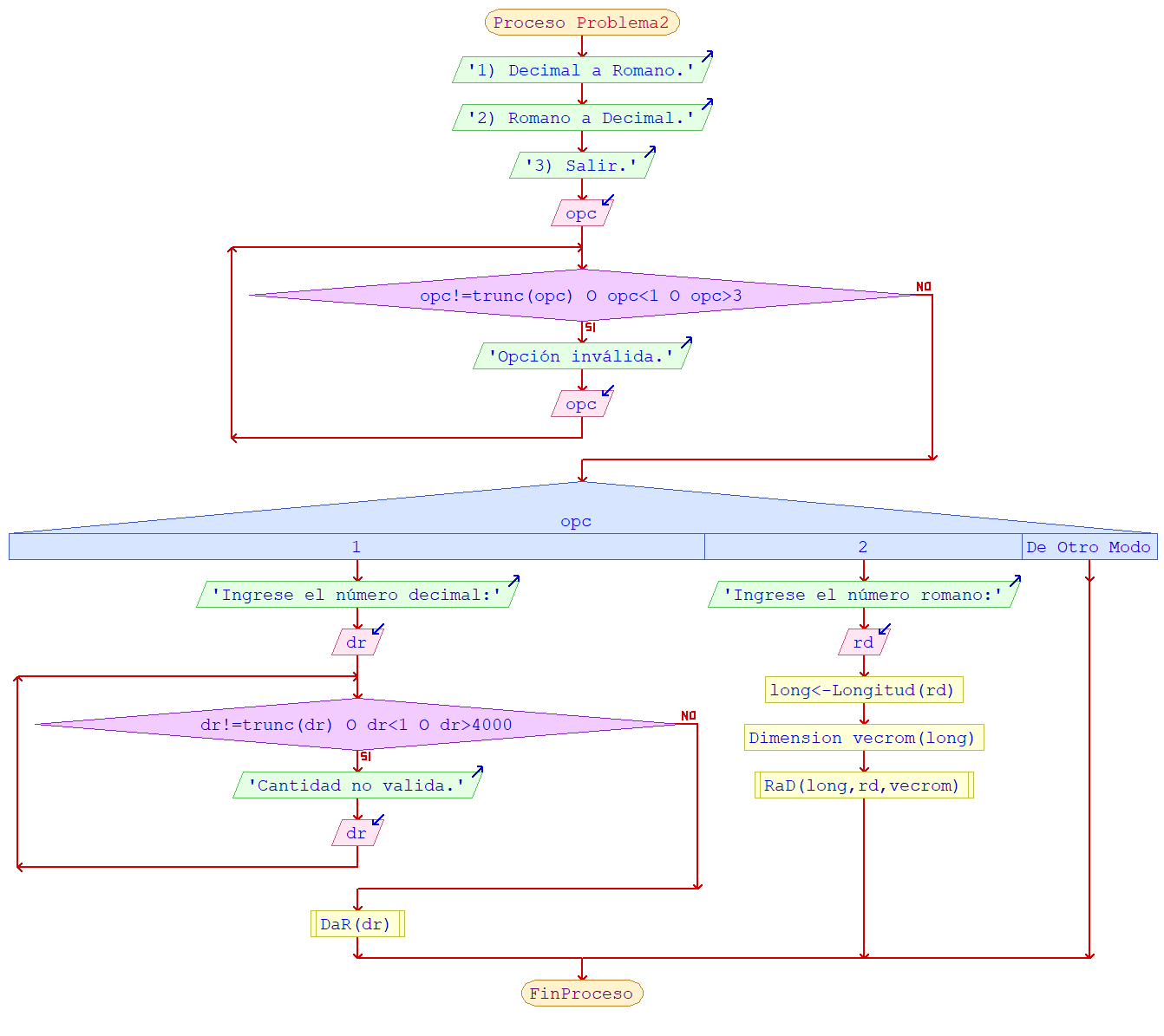
Problema 1: Mi estrategia que aplique en este problema fue hacer una comparación dada la ciudad que pedía y si estaba llenaba un vector auxiliar la cual iba a almacenar todos los caminos que me llevaban a otras ciudades y así poder seguir comparando a que otras ciudades me llevaban.

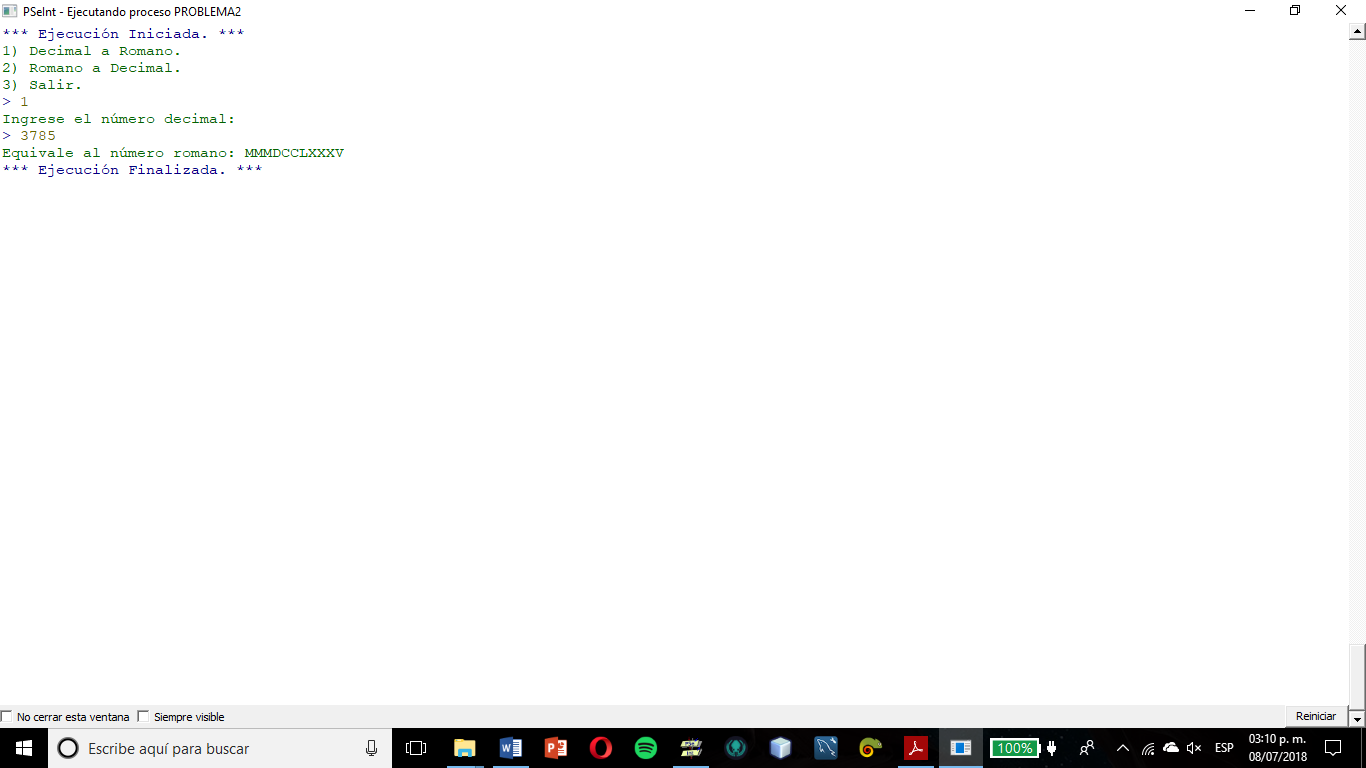
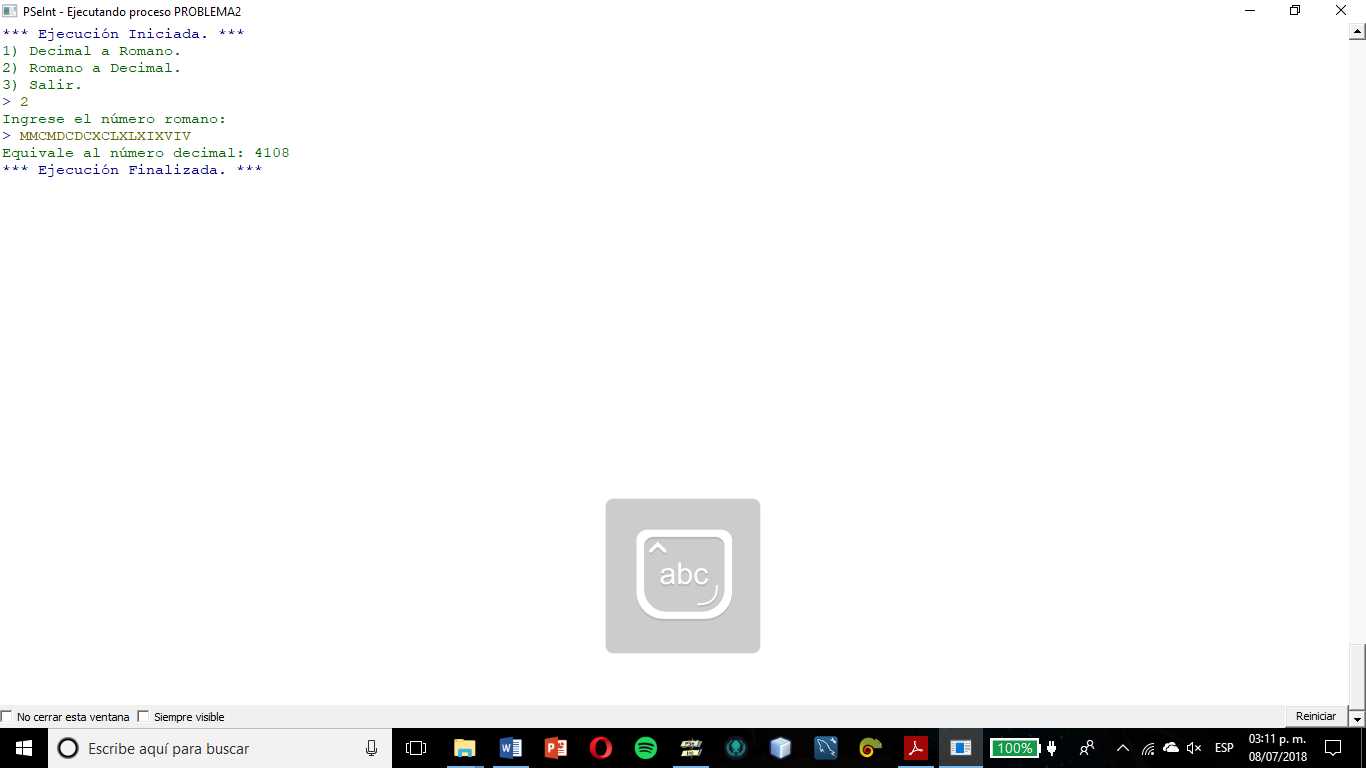
En este problema tuve que utilizar 6 SubProcesos para poder determinar el problema, cada proceso hace distintas cosas.

Corrida:

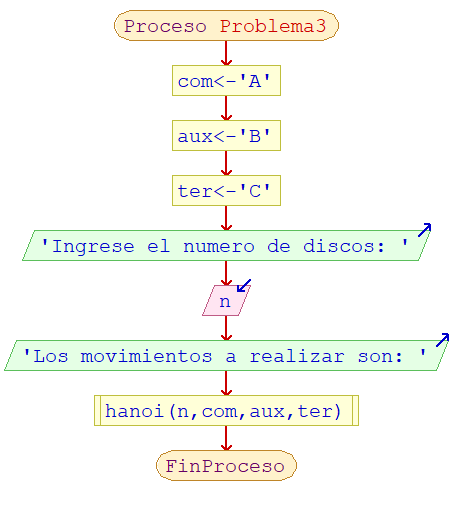


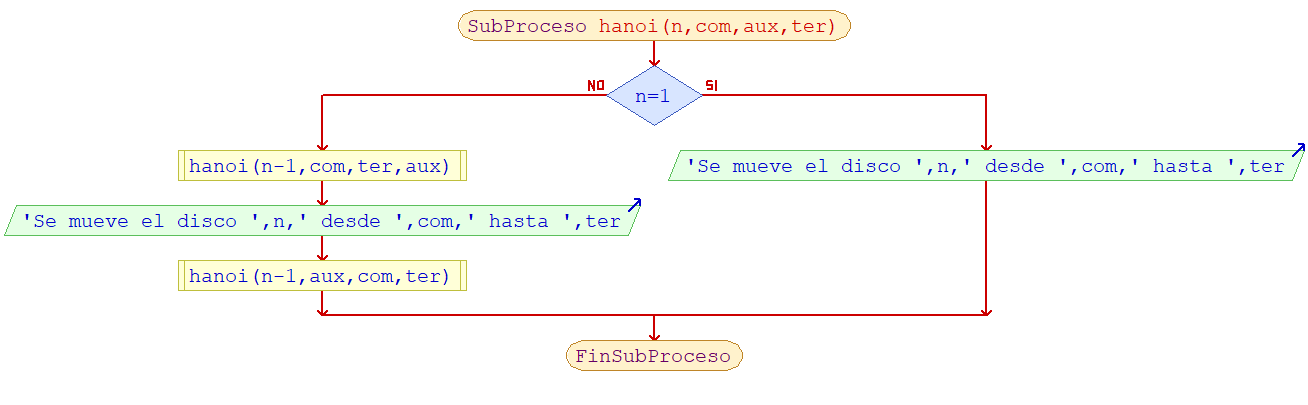
Problema 2: La estrategia que utilice para determinar el número de la transformación de decimal a romano fue que en cuatro vectores vaciar la simbolización de cada número, empezando por unidad, luego decena, seguida de centena y por ultimo millar, intente plantearlo como una matriz, pero vi que para mí seria más fácil manejarlo como vectores unidimensionales, el numero que pasaba por esa función hacia su transformación de determinar cada unidad, decena, centena y millar para luego teniendo ese numero llamar a su expresión correspondiente. Para hacerlo de romano a decimal fue algo mas complicado para mí, ya que tenía que mandarle a la función la cadena, la longitud de la cadena y también un vector, el vector lo utilice como medio de almacenamiento de las subcadenas ya que me acomode mejor teniendo ese vector. Dentro de la función lo que hacía era ya después de meter cada subcadena al vector, recorría ese vector y hacia comparaciones para obtener el símbolo que había ahí para luego, con un acumulador sumarle lo que, equivalida el símbolo, si no se encontraba alguno que corresponda con las denominaciones sale del bucle y menciona el numero romano y la posición de si mismo donde se encuentra el error. Para este problema solo fue necesario utilizar 2 SubProcesos.



Corrida:

Problema 3: En este problema tuve bastante problemas, uno de los cuales empezó en que no sabía como implementarlo, después de varias pruebas logre implementar el modo recursivo, lo cual me llevo a otro problema el cual no había implementado dentro del algoritmo la matriz, estuve verificando e intentado implementar, pero no pude lograrlo correctamente, por lo tanto decidí dejarlo con la forma recursiva, así que la estrategia que sigue en este fue solo adaptar la forma recursiva que me entregaba a la forma de PSeint.





Corrida:

