

Datos del alumno:

Legajo: 1775984

Apellido y nombre: Borrás Facundo

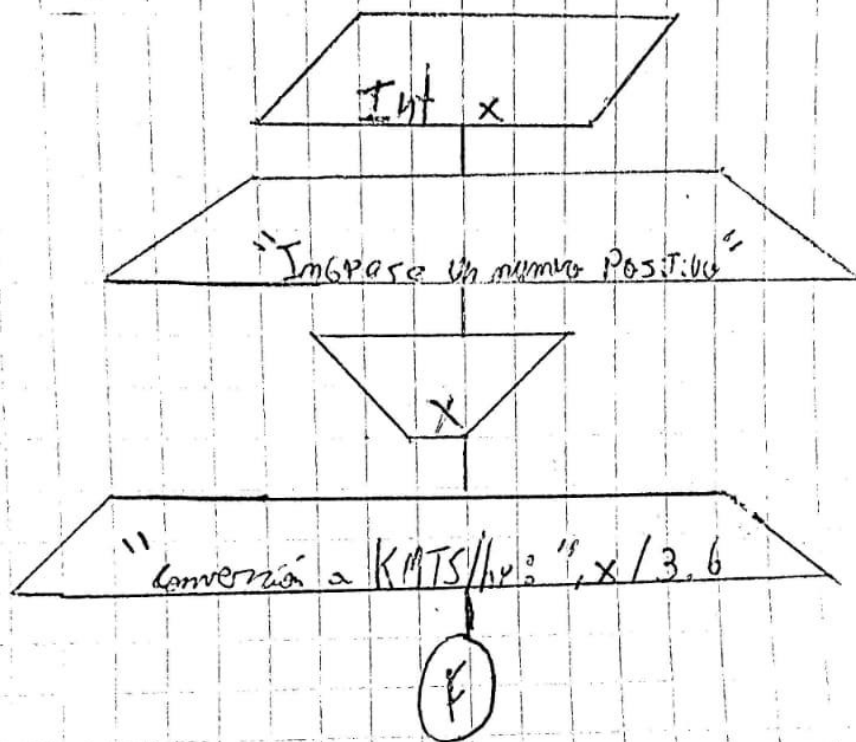
Correo institucional: fborrs@frba.utn.edu.ar

Usuario de gitHub: Facundo Borrás

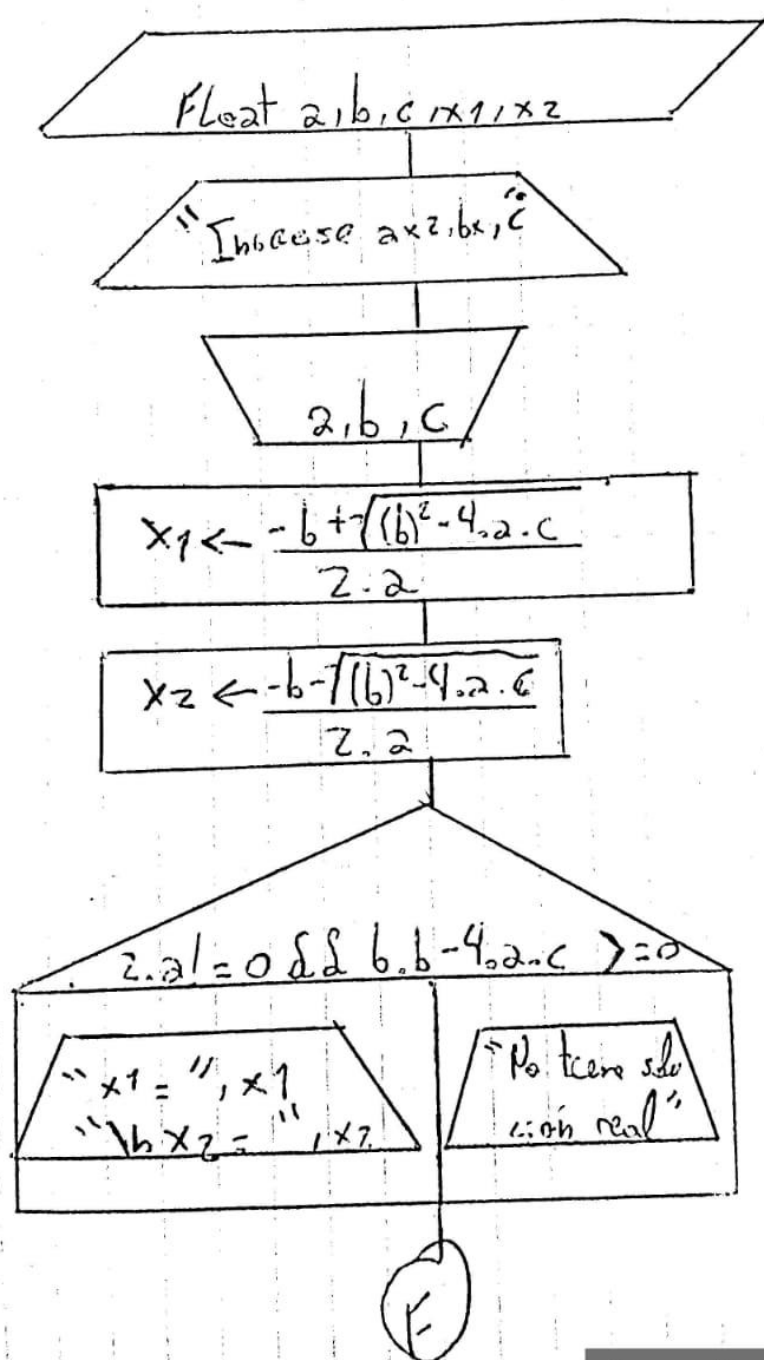
Curso: k1031

Diagramas de Lindsay:

Ejercicio 3_a:



Ejercicio 3_b:



3)_ Análisis del punto a:

Para este punto, hay que tener en cuenta que el objetivo es transformar la velocidad en metros sobre segundos a kilómetros por hora. Sabiendo que la magnitud es la velocidad, esta lleva diferentes características intrínsecas. Ejemplo de esto, si la velocidad es negativa el objeto no se desplaza y no hay velocidad alguna. Por esto, es necesario para lograr el cometido del punto que se ingrese un número positivo para que así sí allá velocidad.

Análisis del punto b:

Partiendo de la base, para poder realizar este ejercicio. Hay que tener en cuenta que hay limitaciones (no usar el `for`, `while`, etc) y, por ende, el resultado se verá afectado. Una vez sabido que se puede utilizar y que no, analizo el problema y como resolverlo. Para ello, es importante verlo primero matemáticamente. La cuadrática da dos soluciones. De las cuales pueden tener distintos valores como: enteros positivos y negativos, enteros con coma, etc. Dentro de este etcétera, están los imaginarios, los cuales hablaré más adelante sobre cómo tratarlos.

En fin, comenzando con el código vuelvo a lo anterior acerca de la matemática. Como hare uso de ella, necesitare funciones matemáticas, para ello agrego la librería *math.h*. Luego elijo las variables que utilizare y el tipo de valor. Las cuales son, las tres que representan a la función cuadrática, y los resultados x_1 y x_2 . Todas las variables utilizare el *float* para poder tener una gama extendida de resultados. Ejemplo, si uso *int* solo podre tener números enteros lo cual me limita. Procedo a seguir escribiendo el código. Primero informo por pantalla que el usuario ingrese los valores necesarios. Luego que este mismo los escriba, mediante el *cin*. Una vez hecho esto, codeo el cómo resolver la cuadrática, donde existen tres posibilidades: factorizar, la resolvente (la que elegí) y la c.c. Hago uso de la función *sqrt*, para así poder resolver la raíz.

Llegados a este punto, ocurren dos posibles caminos. O me da un número que se puede informar por pantalla o uno que no se puede y me salta error. Este número que da error, es el que previamente dije que hablaría. Los números imaginarios. Para ello, como tenemos dos posibles *outcomes*, utilizo el condicional `if` y el `else`. Porque quiero, que imprima por pantalla si se cumple cierta condición; si no se cumple, imprima otro mensaje. Bien, en la condición del `if`, utilizo el conector para dos condiciones. La primera, que el denominador sea distinto de cero. Y la segunda, que la raíz no sea menor a cero. De esta forma evito errores, excepto uno. Que es nuevamente el del número imaginario. Para esto, si aun así persiste, se hace uso del `else` donde se informa por pantalla un mensaje que dice que no tiene solución en el conjunto de los reales. Finalmente, una vez se resuelva, se finaliza el programa.

En conclusión, un ejercicio donde hay que pensar minuciosamente como hacer para poder abarcar la mayor cantidad de resultados por un lado y, por otro, esos casos donde no se puedan resolver.

4: En JavaScript, a diferencia de c/c++, para declarar una variable se utiliza la palabra reservada *var* (que se refiere a variable) y consecutivamente uno elige el nombre de la variable. También, se puede declarar una variable como constante, es decir, que se le tiene que asignar un valor luego de haberla declarada como constante. Y esta misma declaración, mas tarde no se la podrá asignar con ningún otro valor. Otro detalle de este lenguaje es que cuando se declara una nueva variable nueva esta tiene un valor por default que es *undefined* (que la variable nunca fue asignada).

Igualmente, que en c/c++, no se puede utilizar como variable palabras reservadas ni empezar con un número, de esta forma daría en ambos lenguajes error. Continuando con las coincidencias, tampoco esta permitido declarar una misma variable dos veces.

La sentencia if, tanto en JavaScript como en c/c++ es exactamente igual.