

**“UNIVERSIDAD POLITÉCNICA DE PÉNJAMO”**



**Nombre del Alumno:** Brandon Iván Márquez Morales.

**Grado y Grupo:** 1° “A”

**Carrera:** Ingeniería en Software.

**Materia:** Algoritmos.

**Tema:** Sistema de Administración de Almacén de Fertilizantes Químicos S.A.

**Fecha de elaboración:** 20 de octubre del 2020.

**Fecha de terminación:** 25 de octubre del 2020.

**Docente:** Miguel Ángel Saldaña Cabeza.

## Índice

|                    |    |
|--------------------|----|
| Introducción.....  | 1  |
| Marco teórico..... | 2  |
| Desarrollo.....    | 6  |
| Bibliografía.....  | 10 |

## Introducción.

El objetivo del presente documento es analizar y detectar los errores y su resolución de un programa base que trata sobre un sistema de administración de almacén de fertilizantes químicos S. A.

Se toma como base el programa que compartió el profesor Miguel Ángel donde principalmente se ejecutó para que los alumnos observaran el comportamiento del programa posteriormente cada alumno tuvo que haber analizado que y cuantas cosas estaban mal.

Dicho esto, cada alumno debió darle solución a cada uno de los problemas detectados del programa base, principalmente en Pseudocódigo que básicamente es para entenderlo un poco mejor ya que es un lenguaje natural y no hay gran dificultad.

Posteriormente se hace la codificación en cada uno de los lenguajes de programación que se vienen manejando ya que son: C++, Python y Ruby con la finalidad de tener una mejor práctica y cada vez desarrollar mejor nuestras habilidades como programadores, de esta manera entre más práctica halla en cada uno de los alumnos puede aprenderse más aun de memoria su sintaxis y cada una de las instrucciones que se utilizan en dicho lenguaje.

Finalmente, en el desarrollo se explicará claramente de que trata el programa, cuáles fueron y como se resolvieron cada uno de los errores detectados por el alumno Brandon Iván Márquez Morales en general para el Pseudocódigo y la codificación de los tres lenguajes de programación ya mencionados.

## Marco teórico.

### ¿Qué es Python?

Es un lenguaje de programación versátil multiplataforma y multiparadigma que se destaca por su código legible y limpio. Una de las razones de su éxito es que cuenta con una licencia de código abierto que permite su utilización en cualquier escenario.

### ¿Qué es la instrucción “print” en Python?

Una de las acciones básicas e imprescindibles que tiene que realizar un programa es la de mostrar información por pantalla: texto, números, resultados...

Para mostrar texto en Python utiliza la función `print()`, cuya sintaxis es:

```
>>>print('texto') texto
```

```
>>>cadena = "Esto es una cadena"
```

```
>>>print(cadena)
```

Esto es una cadena

### ¿Qué es Ruby?

Ruby es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

### ¿Qué es “puts” en Ruby?

La función `puts` se usa para escribir una secuencia de caracteres al flujo de salida estándar, Además de que este da un salto de línea, Ejemplo:

```
#include <stdio.h> int *puts(const char
```

```
*s);
```

### ¿Qué es C++?

C++ es un lenguaje de programación diseñado en 1979 por Bjarne Stroustrup. La intención de su creación fue extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, C++ es un lenguaje híbrido.

**¿Qué es el tipo de dato “int” en C++?** Este tipo de datos permite trabajar con cualquier número que se encuentre dentro del conjunto numérico de los enteros, por ejemplo 1, 10, 56, 32, etc. Para designar este tipo de datos, se utilizan las siguientes palabras:

- \* int – 16 bits
- \* long – 32 ó 64 bits (dependiendo del compilador y procesador)

**¿Qué es el tipo de dato “float” en C++?**

Este tipo de datos permite trabajar con cualquier número que se encuentre dentro del conjunto numérico de los reales, en otras palabras cualquier número, incluidos los que tienen fracciones decimales, por ejemplo 3.456, 7.231, 65.43, etc. Para designar este tipo de datos, se utilizan las siguientes palabras:

- \* float – 16 bits
- \* double – 32 bits

**¿Qué es una librería?**

En programación, una librería es un archivo o conjunto de archivos que se utilizan para facilitar la programación. Las librerías, también llamadas "frameworks", consisten en archivos de código a los que llamamos al principio de la página, por ejemplo, una librería JavaScript será un archivo en JavaScript que insertamos al principio de la página.

**¿Para que sirve la librería #include<iostream> en C++?**

Esta librería sirve para Entrada y Salida de datos en C++ i: input, o: output, stream: corriente.

**¿Para qué sirve “using namespace std;” en C++?**

Instrucción para indicar al programa que se usará de manera estándar el flujo de entrada y salida de datos y poner: cout<< y cin>> sin estar que poniendo siempre “std : :” antes de ellos.

**¿Qué es y para que sirve “int main() {}” en C++?** Es la función principal de C++ que sirve para poner dentro todo el código a desarrollar.

**¿Qué es y para qué sirve “return 0;” en C++?** Retorno para que el programa detecte que ha finalizado todo correctamente.

**¿Para qué sirve la instrucción “cout<<” en C++?**

La instrucción `cout` es utilizada como método de salida de datos por pantalla (realiza una impresión en pantalla). La sintaxis de uso de esta instrucción (forma de escribir correctamente la instrucción) es la siguiente:

```
cout<<"Mensaje a imprimir";
```

### ¿Para qué sirve la instrucción “`cin>>`” en C++?

La instrucción `cin`, es utilizada para la entrada de datos por medio del teclado. La sintaxis de uso de esta instrucción es la siguiente: `cin>>NombreDeLaVariable;`

### ¿Qué es PSeInt?

PSeInt es la abreviatura de los estados de computación de PSeudo Intérprete, una herramienta educativa creada en Argentina, utilizada principalmente por estudiantes para aprender los fundamentos de la programación y el desarrollo de la lógica. Es un software muy popular de su tipo y es ampliamente utilizado en universidades de Latinoamérica y España.

Utiliza pseudocódigo para la solución de algoritmos.

### ¿Qué es un ciclo?

Un bucle o ciclo, en programación, es una secuencia que ejecuta repetidas veces por una instrucción/un trozo de código, hasta que la condición asignada a dicho bucle deja de cumplirse. Los tres bucles más utilizados en programación son el bucle `while`, el bucle `for` y el bucle `do-while`.

### ¿Qué es una variable?

Una variable es donde se guarda (y se recupera) datos que se utilizan en un programa.

Cuando escribimos código, las variables se utilizan para:

Guardar datos y estados.

Asignar valores de una variable a otra.

Representar valores dentro de una expresión matemática.

Mostrar valores por pantalla.

Todas las variables deben ser de un tipo de datos, ya sea un dato de tipo primitivo, como un número o texto, o un dato abstracto, como un objeto que se ha creado.

Lo habitual es que haya una serie de restricciones. La mayoría de los lenguajes de programación tienen estas condiciones:

Pueden ser letras, números y el símbolo `_`. Con estos límites: no puede tener espacios. no empezar con un número.

no puede ser una palabra reservada. Por ejemplo, `if`, `for`, `while`...

Algunos lenguajes pueden permitir empezar con carácter especial, como \$.

### **¿Qué son los datos de entrada y salida?**

En un programa, los datos de entrada son los que la computadora va a procesar. Los datos de salida son datos derivados, es decir, obtenidos a partir de los datos de entrada. Por esta razón, a los datos de salida se les considera más significativos que a los datos de entrada. Ambos tipos de datos son información (textos, imágenes, sonidos, vídeos,...) que maneja la computadora. Sin embargo, en un sentido más filosófico, a los datos de entrada se les considera la materia prima de los datos de salida, considerados estos como la verdadera información. **¿Qué es una cadena de caracteres?**

Una cadena también llamada "string" (así es más común porque casi todos los lenguajes de programación son en inglés son un tipo de datos que representan texto. Se llaman cadenas porque se componen de caracteres únicos encadenados, es decir ligados entre sí. Pueden ser asignadas a una variable (dato introducido por el usuario del cual el valor o descripción puede cambiar con cada ejecución del programa) o combinarlas.

### **¿Qué es una operación en programación?**

Una vez que se comprende el concepto de variable y/o constante, sus correspondientes tipos de datos y las expresiones que pueden escribirse al combinar operadores y operandos es posible realizar operaciones, es decir acciones que conduzcan a disponer de variables con valores que pueden obtenerse ya sea por parte de los usuarios de los programas o a su vez como resultado de la ejecución de una expresión.

## Desarrollo.

### **Principalmente se explicará de que trata el programa:**

Es un sistema de administración de almacén de fertilizantes químicos S. A. que maneja tres productos los cuáles son: Sulfato de amonio, Urea y Fosfato manejándolas en toneladas, este programa o algoritmo cuenta con un menú principal de tiene 5 tareas a ejecutar las cuales son:

- 1.- Registro de existencia inicial.
- 2.- Entrada de mercancía.
- 3.- Salida de mercancía.
- 4.- Reporte de existencia.
- 5.- Salir del sistema de administración de almacén de fertilizantes químicos S. A.

### **1.- Registro de existencia inicial.**

Para poder ejecutar cualquier otra tarea del menú a excepción del salir del sistema, se debe de realizar primero que nada el registro de existencia inicial para cada uno de los tres productos, además no se debe de permitir registrar cantidades negativas de ser así no habrá registro de existencia inicial y no se podrá ejecutar cualquier otra tarea.

Una vez cumplido con lo que pide, el registro de existencia inicial no se puede volver a dar otro registro de existencia inicial.

### **2.- Entrada de mercancía.**

Al ingresar a esta opción se debe desplazar otro menú para decidir a cuál producto se le quiere dar entrada de mercancía.

Se puede dar entradas de mercancía tanto en enteros como decimales manejando la unidad en toneladas, No debe permitir entrar cantidades negativas de ser así no hay entrada de mercancía.



### **3.- Salida de mercancía.**

Al ingresar a esta opción se debe desplazar otro menú para decidir a cuál producto se le quiere dar salida de mercancía.

Se puede dar salidas de mercancía tanto en enteros como decimales manejando la unidad en toneladas, No debe permitir salir cantidades negativas de ser así no hay salida de mercancía.

Además, si se pretende retirar más mercancía de la que no se cuenta, no se podrá retirar mercancía ya que no se cuenta con esa cantidad.

### **4.- Reporte de existencia.**

Al ingresar a esta opción se muestra la existencia actual de cada uno de los tres productos en toneladas.

### **5.- Salir del sistema de administración de almacén de fertilizantes químicos S. A.**

Esta opción permite finalizar la ejecución del programa.

**A continuación, se mostrará la lista de errores detectados:**

- 1.- El permitir hacer entradas y salidas de mercancías sin haber hecho el registro de existencia inicial para el Sulfato de Amonio, Urea y Fosfato.
- 2.- Permitir registrar más de una vez el registro de existencial para el Sulfato de Amonio, Urea y Fosfato.
- 3.- En las entradas y salidas de mercancía, permitir registrar cantidades negativas
- 4.- En las entradas y salidas de mercancía, permitir que a fuerzas se tenga que darle entrada y salida de mercancía a los 3 productos (Sulfato de Amonio, Urea y Fosfato).
- 5.- En las salidas de mercancía, permitir retirar más mercancía de la que no se cuenta
- 6.- Y finalmente algo adicional que yo creo que podría ser un error es no tener seguridad en un sistema de administración de almacén de fertilizantes químicos.

**Solución a los errores detectados:**

1.- Primero ya se debe de tener inicializada cada una de las variables de existencia inicial para cada uno de los elementos (Sulfato de Amonio, Urea y Fosfato) una vez echo esto al querer ingresar tanto para las entradas y salidas de mercancía lo primero que se debe hacer es una comparación con la sentencia “if” preguntando si cada una de las 3 variables de existencia inicial son iguales a cero, si es así entonces se imprime un mensaje de salida indicando que “No se puede dar entrada o salida de mercancía ya que no se han hecho los registros de existencia inicial para Sulfato de Amonio, Urea y Fosfato” y caso contrario, es decir que las variables de existencia inicial sean mayores o igual a uno entonces se ejecutan las operaciones correspondientes.

2.- Para evitar que no se hagan más de un registro de existencia inicial para Sulfato de Amonio, Urea y Fosfato, se utiliza la sentencia “if” preguntado si su existencia inicial es mayor o igual a uno, si se cumple esa condición quiere decir que ya se le otorgo un valor a cada una de las variables de existencia inicial por lo tanto no permitirá que deje hacer

un segundo registro de existencia inicial posteriormente si eso se cumple se imprime un mensaje en pantalla indicando que “No puede hacer más de una vez el registro de existencia inicial para Sulfato de Amonio, Urea y Fosfato”, caso contrario se ejecutan las operaciones correspondientes.

**3.-** Para evitar que deje ingresar o retirar cantidades negativas, se compara si la cantidad es menor o igual a menos uno con la sentencia “if”, si eso se cumple se imprimirá mensaje de salida indicando que “No puede ingresar cantidades negativas, por favor retire una cantidad positiva” e inmediatamente nos retorna a volvernos a pedir la cantidad a ingresar o retirar y darle lectura a la cantidad ingresada esto con el ciclo “while”

**4.-** Se debe hacer un menú para indicar a cuál de los tres productos se le quiere dar entrada o salida de mercancía, una vez echo el menú se utiliza la sentencia “case” en este caso solo para Pseint, C++ y Ruby, en Python se utiliza “if” y “elif”, según la opción seleccionada se hará la operación para dicho producto.

**5.-** Para esta solución se hace una comparación con sentencia “if” donde la lectura es: “si la cantidad que se pretende retirar es mayor a la existencia actual de dicho producto” entonces se imprime mensaje diciendo que: “No puede retirar más cantidad ya cuenta con esa cantidad que se quiere retirar” y en caso contrario si es que tampoco se ingresó una cantidad negativa a retirar se imprime mensaje diciendo que: “La salida ha sido exitosa”

**6.-** Se hizo un Login utilizando la sentencia “if” comparando si los atributos “usuario” y “contraseña” sean igual ha “aa” y “aa” si se cumple entonces se accede a todo el sistema de administración de almacén de fertilizantes químicos S. A, Caso contrario se imprime mensaje “USUARIO O CONTRASEÑA INCORRECTOS”

## Bibliografía

- ClubEnsayos. (2014). *ClubEnsayos*. Obtenido de ClubEnsayos: <https://www.clubensayos.com/Tecnolog%C3%ADa/Caracteristicas-De-La-Programacion-Estructurada/1553103.html#:~:text=%20Caracteristicas%20De%20La%20Programacion%20Estructurada%20%201,de%20ciclos%29%3A%20Su%20construcci%C3%B3n%20tambi%C3%A9n%20hace...%20More%2>
- GUI. (2018). *GUI*. Obtenido de GUI: <http://www2.elo.utfsm.cl/~iwg101/ClaseVisual.pdf>
- lifeder.com. (2015). *lifeder.com*. Obtenido de lifeder.com: <https://www.lifeder.com/programacion-orientada-a-eventos/>
- not, C. o. (2018). *Coding or not*. Obtenido de Coding or not: <https://codingornot.com/que-es-la-programacion-orientada-a-aspectos-aop>
- OkHosting. (2018). *OkHosting*. Obtenido de OkHosting: <https://okhosting.com/blog/paradigma-de-programacion/>