|  |  |  |
| --- | --- | --- |
| Imagen que contiene dibujo, taza  Descripción generada automáticamente | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Ingeniería Karina García Morales |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 1120 |
| *No de Práctica:* | Pseudocódigo  *5º* |
| *Integrante:* | Yushua Haza Jibaja |
| *No. de Equipo de cómputo empleado:* |  |
| *No. de Lista o Brigada:* | 22 |
| *Semestre:* | 2022 - 1 |
| *Fecha de entrega:* | 28 de septiembre de 2021 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Objetivo:**

El alumno elaborará pseudocódigos que representen soluciones algorítmicas empleando

la sintaxis y semántica adecuadas.

**Actividades:**

• Elaborar un pseudocódigo que represente la solución algorítmica de un

problema en el cual requiera el uso de la estructura de control de flujo

condicional.

• A través de un pseudocódigo, representar la solución algorítmica de un

problema en el cual requiera el uso de la estructura de control iterativa.

**Desarrollo de la práctica**

**Sintaxis de pseudocódigo**

El lenguaje pseudocódigo tiene diversas reglas, las más importantes son:

1. Alcance: Se refiere a que todo algoritmo debe tener un Inicio y un Fin, esto es porque una vez que se haya solucionado el problema gracias al algoritmo, no se debe mantener a este en operación, puesto que no hay problema o solicitud que atender
2. Palabras reservadas con mayúsculas: Son palabras reservadas del programa en cuestión que se deben escribir con mayúsculas
3. Sangría o tabulación: El pseudocódigo debe tener diversas alineaciones para que el código sea más fácil de entender y depurar.
4. Lectura / escritura: Para mostrad lectura de datos se utiliza LEER. Para mostrar escritura de información se utiliza ESCRIBIR.

Ejemplo

Leer num

Escribir “Buen intento, reintenta de nuevo”

1. Declaración de variables: la declaración de variables la definen un identificador (nombre), seguido de dos puntos, seguido del tipo de dato, es decir: <nombreVariable>:<tipoDeDato>

Ejemplo

Definir num Como Entero

Contador: ENTERO

Resultado: REAL

1. Operadores aritméticos: Se tiene la posibilidad de utilizar operadores aritméticos y lógicos: Operadores aritméticos: suma (+), resta (-), multiplicación (\*), división (/), división entera (div) esto ya es en programación hay que omitirla, módulo (mod), exponenciación (^), asignación (:=). Operadores lógicos: igualdad (=), Y-lógica o AND (&), O-lógica u OR (|), negación o NOT (!), relaciones de orden (<, >, <=, >=) y diferente (<>).

Estructuras de control de flujo

Las estructuras de control de flujo son las que indican el orden de ejecuón de las estructuras y la repetición de estas mismas estructuras

Existen 3, siendo estas:

-Secuencial

-Condicional

-Repetitiva o Iterativa

Estructura secuencial

Es aquella que sigue un flujo continuo directo sin bifurcación alguna, es decir, que no se detiene según ninguna estructura posterior, es un algoritmo que no se repite ni tiene ciclos internos

Estructura condicional

Es aquella que en al menos una ocasión se ingresa un comando condicional que bifurca el resultado o condiciona este mismo según sea el dato ingresado que pase por el mismo condicional

Estructura iterativa

Es aquella que permite la ejecución de un bloque de instrucciones dentro de un ciclo MIENTRAS, es decir, que solo en tanto el dato ingresado sea admitido por el mientras, estas instrucciones se estarán ejecutando

El comando MIENTRAS valida primero la condición y si esta es verdadera permite la ejecución del bloque de instrucciones

Ejemplo

MIENTRAS condición Entonces

(ACCIÓN

FIN MIENTRAS

**Funciones**

Cuando un problema es muy complejo de resolver vía algoritmo, se pueden emplear bloques que resuelvan partes concretas del problema, así diversificando las tareas y llegando a una solución acertada, así también facilita en algunas que en las ocasiones donde el resultado no es acertado, se puede encontrar con mayor facilidad el problema gracias a que el algoritmo esta desarrollado por bloques, a estos bloques se les llaman métodos o funciones

Una función esta formada por un identificador de función (nombre), de 0 a n parámetros de entrada y un valor de retorno:

INICIO

FUNC identificador (var:TipoDato,…,var:TipoDato) RET: TipoDato

(ACCIONES

FIN FUNC

FIN

**Ejercicios en clase**

**1-Elaborar un pseudocódigo que represente la solución algorítmica de n problema en el cual requiera el uso de la estructura de control de flujo condicional**

**2-A través de un pseudocódigo, representa la solución algorítmica de un problema en el cual requiera uso de la estructura de control iterativa**

**Ejercicio 0**

AlumnoHaza: REG

numeroCuenta:Entero

carrera:CADENA

nombreAlumno:CADENA

promedioAlumno:REAL

FIN REG

**Ejercicio 1**

Inicio

Definir peli como entero

Escribir “Elige una película según el número”

Escribir “1) Titanes del Pacífico

Escribir “2) Transformers”

Escribir ”3) Robots”

Leer peli

Si peli=1 Entonces

Escribir “La película será Titanes del Pacífico”

SiNo

Si peli=2 Entonces

Escribir “La película será Transformers”

SiNo

Si peli=3 Entonces

Escribir “La película será Robots”

FinSi

FinSi

FinSi

Fin

|  |  |  |
| --- | --- | --- |
| Iteración | Valor ingresado | Dato de salida |
| 1 | 1 | Titanes del Pacífico |
| 2 | 2 | Transformers |
| 3 | 3 | Robots |

Algoritmo YushuaHazaAct5

Inicio

Definir Estado Como Entero

Escribir 'Ingresa tus calificaciones'

Leer Estado

Si Estado>=6 Entonces

Escribir "Felicidades, has aprobado"

SiNo

Escribir "Vas reprobando, tienes que corregir tus tareas"

FinSi

FinAlgoritmo

|  |  |  |
| --- | --- | --- |
| Iteración | Valor de entrada | Valor de salida |
| 1 | 5 | Reprobado |
| 2 | 10 | Aprobado |
| 3 | 7 | Aprobado |

**Intento 2**

Algoritmo YushuaHazaAct5

Definir Estado Como Entero

Escribir 'Ingresa tus calificaciones'

Leer Estado

Si Estado=6 Entonces

Escribir 'Felicidades, has aprobado'

SiNo

Si Estado<10 Entonces

Escribir 'Vas reprobando, tienes que corregir tus tareas'

SiNo

Escribir "Felicidades, has aprobado"

FinSi

FinSi

FinAlgoritmo

|  |  |  |
| --- | --- | --- |
| Iteración | Valor de entrada | Valor de salida |
| 1 | 0 | Reprobado |
| 2 | 7 | Aprobado |
| 3 | 5 | Reprobado |

**Intento incorrecto por estructura de condición**

Algoritmo YushuaHaza

Definir MATa,MATb,MATc Como Entero

Definir Estado Como Real

Escribir 'Ingresa tus calificaciones'

Leer MATa,MATb,MATc

Estado <- ((MATa+MATb+MATc)/3)

Si Estado>9 Entonces

Escribir 'Tu promedio es',(MATa+MATb+MATc)/3,'Y estas acutalmente aprobado'

SiNo

Si 6<Estado>8 Entonces

Escribir "Es un alumno regular"

SiNo

Escribir "Es un alumno reprobado"

FinSi

FinSi

Escribir 'Tu promedio es',(MATa+MATb+MATc)/3

FinAlgoritmo

**Conclusión**

Se pudieron llegar a los aprendizajes deseados, los cuales cumplen con los objetivo propuesto por la práctica, el alumno efectivamente pudo aprender la sintaxis del pseudocódigo, así como aprendió a desarrollarlo para llegar a desarrollar algoritmos que resuelven problemas aritméticos simples

**Tarea:**

Realizar el pseudocódigo de los siguientes ejercicios, agregar análisis y pruebas de escritorio

1.- Algoritmo que valide la calificación que ingrese el usuario, esta debe estar entre 5 y 10. Indicar si ha aprobado con calificación mayor a 6; un letrero "Aprobado, felicidades", en caso de no aprobar "Reprobado, tienes una nueva oportunidad"

2.- Del ejercicio de clase (el menú) modifica el ciclo hacer mientras, en donde valide, si el usuario no elige las opciones 1ó 2 para caracteres deben ser 'a', 'b' o 'c') imprimir un letrero de "error"(es el default) y volver a mostrarle el menú.

3.-Algoritmo que muestre la numeración del 1 al 1000, emplea la estructura MIENTRAS, HACER MIENTRAS Y PARA

4.- Solicitar al usuario que ingrese la variable, si ingresa una variable diferente a la letra 'a' o 'A', volver a solicitarla, en caso de contrario, imprimir la letra 'a' o 'A' que ingresó el usuario.

Nota: No olviden la caratula, conclusiones y su biografía

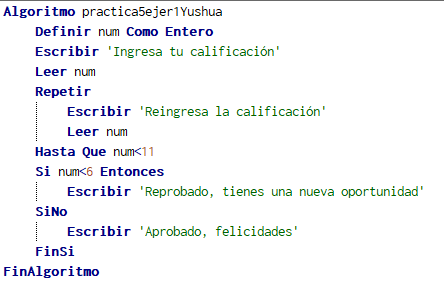
**1.- Algoritmo que valide la calificación que ingrese el usuario, esta debe estar entre 5 y 10. Indicar si ha aprobado con calificación mayor a 6;un letrero "Aprobado, felicidades", en caso de no aprobar "Reprobado, tienes una nueva oportunidad"**

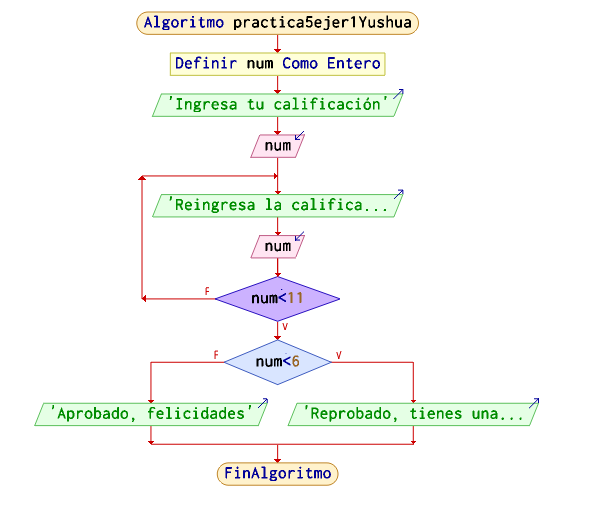
**Análisis**

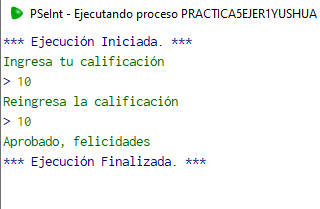
**Dato de entrada:** una calificación númerica

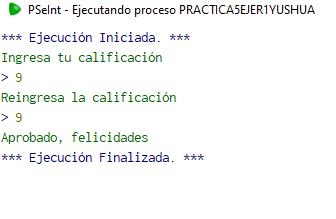
**Restricción:** dadas las características que debe tener el programa, no se puede ingresar una cantidad decimal, solo entera no mayor a 11, puesto que sí se puede tener una calificación de 10 en un sistema base 10

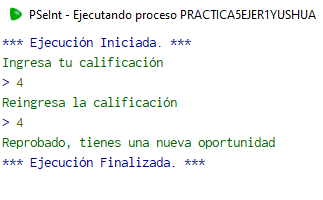
**Dato de salida:** impresión de letrero de “Aprobado…” o “Reprobado…”





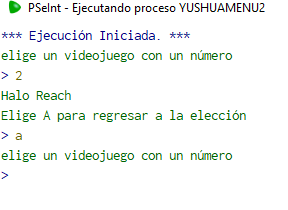


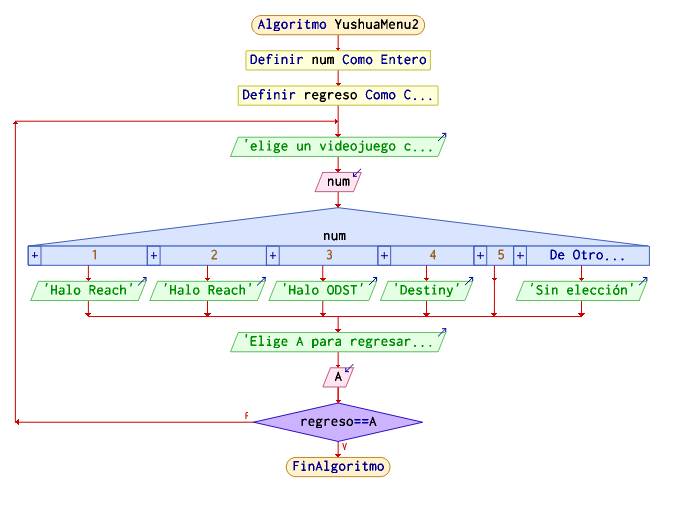


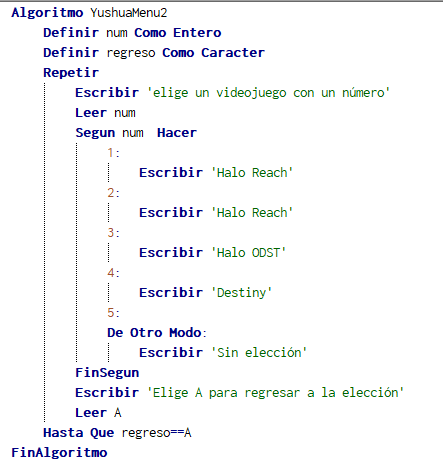


|  |  |  |
| --- | --- | --- |
| **Iteración** | **Calificación** | **Dato de salida** |
| 1 | 10 | Aprobado |
| 2 | 9 | Aprobado |
| 3 | 4 | Reprobado |

**2.- Del ejercicio de clase (el menú) agregar el ciclo hacer mientras, en donde valide, si el usuario no elige las opciones 1,2 ó 3(para carácter pueden ser 'a', 'b' o 'c') imprimir un letrero de "error"(es el default) y volver a mostrarle el menú.**







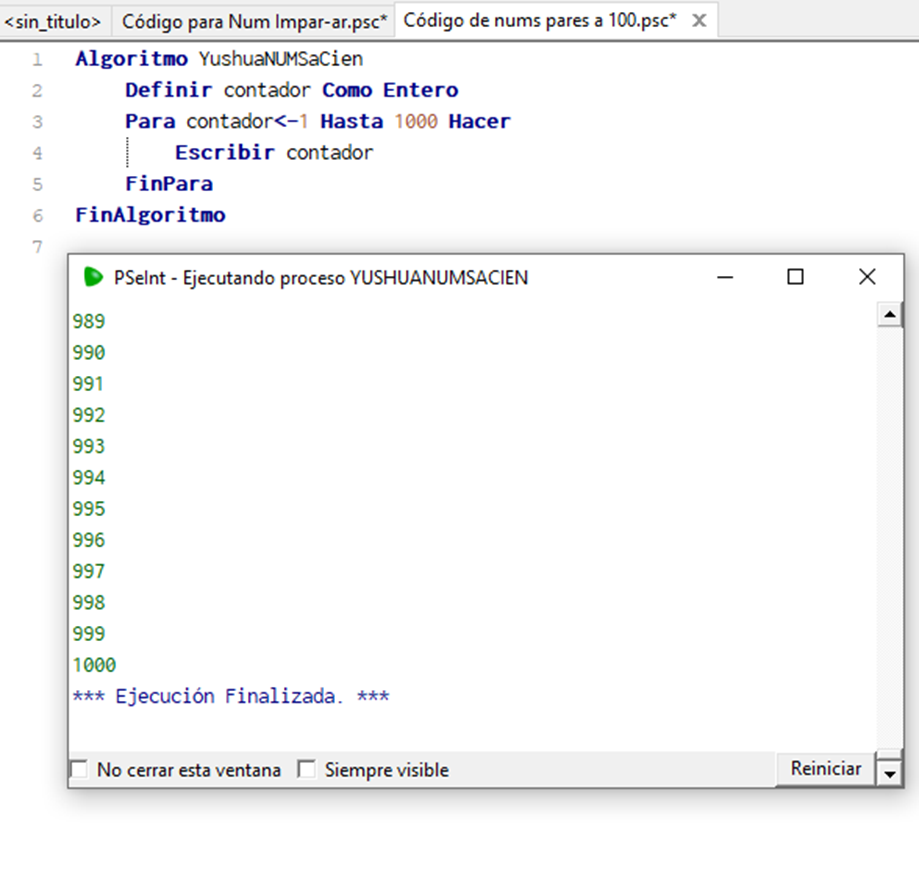
**3- Realiza un diagrama que muestre l numeración del 1 al 1000, emplea la estructura MIENTRAS y realizarlo también con la estructura PARA**

**Analisis**

**Datos de entrada:** El algoritmo debe arrojar la numeración de números hasta el valord de 1000

**Restricción:** imprimir hasta valor de 1000

Dato de salida: impresión de números hasta el valor de 1000



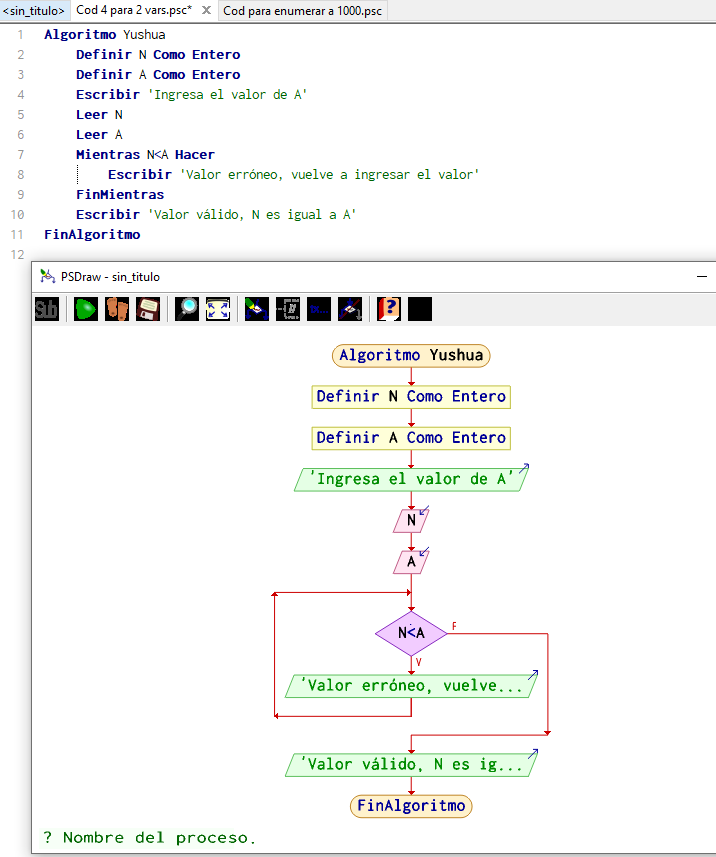
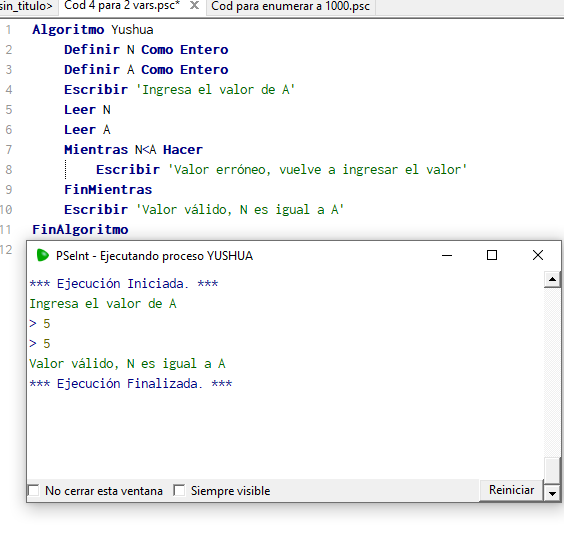
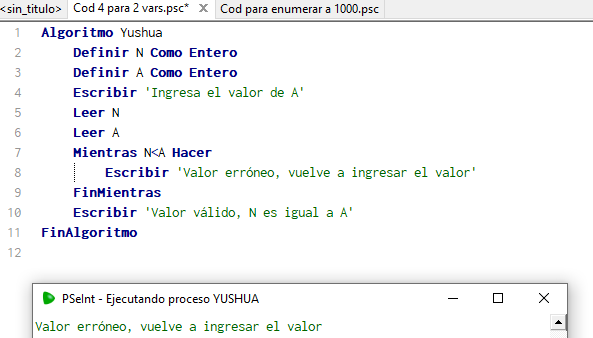
**4- Solicitar al usuario que ingrese la variable, si ingresa una variable diferente a la letra 'a' o 'A', volver a solicitarla, en caso de contrario, imprimir la letra 'a' o 'A' que ingresó el usuario.**

Analisis:

Datos de entrada: valor de A, valor de N, N y A tienen que ser identicos de valor para poder imprimir la solución

Restricción: la misma dada por el problema, N y a tienen que ser identicos de valor

Dato de salida: Impresión “Valor válido, N es igual a A

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración** | **Valor 1** | **Valor 2** | **Dato de salida** |
| 1 | 5 | 5 | Igualdad |
| 2 | 5 | 6 | Desigualdad |
| 3 | 4 | 4 | igualdad |

Link de Github:

# Referencias

Gálvez, J. A., & García Cano, E. E. (6 de abril de 2018). *Guía práctica de estudio 05: Pseudocódigo*. Obtenido de http://lcp02.fi-b.unam.mx/: http://lcp02.fi-b.unam.mx/