

# PRUEBA 1

PROGRAMACIÓN - TICS 100

21-04-2018

## Importante.

Lee detenidamente los problemas. Un formulario con las instrucciones básicas de programación se encuentra adjunto a esta prueba. **Tiempo Máximo 120 minutos.** Apaga su celular. **Utiliza SOLO las hojas en blanco para responder las preguntas. No contestes sobre el mismo enunciado.** Puedes emplear lápiz grafito pero debes cargarlo bien.

### PROBLEMA 1 (20 puntos)

Un acertijo que se les plantea a los niños cuando están aprendiendo lógica matemática es el de la "rana en el pozo". En este acertijo los niños deben responder cuántos días tarda una rana en salir de un pozo de cierta profundidad, dado que de día avanza una determinada cantidad de metros y por la noche retrocede una cierta cantidad de metros.

Por ejemplo, supongamos que el pozo tiene 31 metros de profundidad, la rana avanza 5 metros durante el día y desciende 2 metros por la noche. Supongamos además que la rana inicia sus intentos de día. En tal caso, la rana demorará 10 días en salir. En la Tabla 1 te mostramos el detalle del avance diario de la rana con estos datos de entrada (profundidad = 31, avance=5, retroceso=2).

Tabla 1. Ejemplo de avance y retroceso

Día	Avanza	Retrocede	Situación Final
1	5	2	3
2	5	2	6
3	5	2	9
4	5	2	12
5	5	2	15
6	5	2	18
7	5	2	21
8	5	2	24
9	5	2	27
10	5		Sale del pozo

Ahora, te pedimos que crees el algoritmo que permite resolver el problema de forma genérica. Es decir, determinar cuántos días demora la rana en salir de un pozo para cualquier combinación de valores para la profundidad, el avance y el retroceso (en metros). Supón que la rana siempre inicia sus esfuerzos de día.

### PARTE A)

Escribe el algoritmo en pseudo-código o el diagrama de flujo. El algoritmo debe :

- (1pts) **Solicitar** al usuario ingresar la profundidad en metros del pozo
- (1pts) **Solicitar** al usuario ingresar la cantidad de metros que la rana avanza de día
- (1pts) **Solicitar** al usuario ingresar la cantidad de metros que la rana cae de noche
- (11 pts) **Calcular** el número de días que demora la rana en salir y **desplegar** esta información por pantalla

En la Figura 1 se muestra un ejemplo de cómo debe ser la interacción del algoritmo con el usuario.

**Fig. 1.** Ejemplo de ejecución de este algoritmo (parte A)

```
*** Ejecución Iniciada. ***  
Ingrese la altura en metros del pozo (en metros)  
> 31  
Ingrese la cantidad de metros que avanza de día (en metros)  
> 5  
Ingre la cantidad de metros que cae de noche (en metros)  
> 2  
La rana se demora 10 días en salir  
*** Ejecución Finalizada. ***
```

Solución:

a)

Algoritmo rana\_en\_pozo

Mostrar "Ingrese la altura en metros del pozo (en metros)"

Leer altura

Mostrar "Ingrese la cantidad de metros que avanza de día (en metros)"

Leer avanza

Mostrar "Ingrese la cantidad de metros que cae de noche (en metros)"

Leer retrocede

dia = 0

final = 0

Mientras final < altura Hacer

    dia = dia + 1

    final = final + avanza

    Si final < altura Entonces

        final = final - retrocede

    FinSi

FinMientras

Mostrar "La rana se demora ", dia, " días en salir"

FinAlgoritmo

## PARTE B)

Modifica tu algoritmo (el que propusiste en la parte A) de forma que ahora además verifique si los datos ingresados son válidos. Los datos son válidos si cumplen las siguientes condiciones:

- La profundidad del pozo debe ser mayor a cero
- La cantidad de metros que avanza la rana durante el día debe ser mayor a cero
- La cantidad de metros que la rana retrocede en la noche debe ser mayor o igual a cero
- La cantidad de metros que la rana avanza en el día debe ser mayor que la cantidad de metros que retrocede en la noche

Es decir, el nuevo algoritmo debe verificar que:

- (1pts) La profundidad del pozo sea un valor válido
- (1pts) La cantidad de metros de avance sea un valor válido
- (1pts) La cantidad de metros de retroceso sea un valor válido
- (1pts) La relación entre avance y retroceso sea válida
- (2pts) Si alguno de los datos anteriores no es válido el algoritmo debe terminar, sin realizar el cálculo, con el siguiente mensaje: "Dato inválido. No puedo calcular la cantidad de días".

**Fig. 2.** Ejemplo de ejecución de este algoritmo (parte B)

```
*** Ejecución Iniciada. ***  
Ingrese la altura en metros del pozo (en metros)  
> 30  
Ingrese la cantidad de metros que avanza de día (en metros)  
> 5  
Ingrese la cantidad de metros que cae de noche (en metros)  
> 7  
Dato inválido. No puedo calcular la cantidad de días  
*** Ejecución Finalizada. ***
```

Solución b)

Alternativa 1: comprobación después de capturar todos los datos

Algoritmo rana\_en\_pozo

Mostrar "Ingrese la altura en metros del pozo (en metros)"

Leer altura

Mostrar "Ingrese la cantidad de metros que avanza de día (en metros)"

Leer avanza

Mostrar "Ingrese la cantidad de metros que cae de noche (en metros)"

Leer retrocede

Si altura > 0 Y avanza > 0 Y retrocede >= 0 Y avanza > retrocede Entonces

    dia = 0

    final = 0

    Mientras final < altura Hacer

        dia = dia + 1

        final = final + avanza

        Si final < altura Entonces

            final = final -retrocede

        FinSi

    FinMientras

    Mostrar "La rana se demora ", dia, " días en salir"

SiNo

    Mostrar "Dato inválido. No puedo calcular la cantidad de días"

FinSi

FinAlgoritmo

Alternativa 2: comprobación después de cada dato

Algoritmo rana\_en\_pozo

Mostrar "Ingrese la altura en metros del pozo (en metros)"

Leer altura

Si altura > 0 Entonces

Mostrar "Ingrese la cantidad de metros que avanza de día (en metros)"

Leer avanza

Si avanza > 0 Entonces

Mostrar "Ingrese la cantidad de metros que cae de noche (en metros)"

Leer retrocede

Si retrocede >= 0 Y avanza > retrocede Entonces

dia = 0

final = 0

Mientras final < altura Hacer

dia = dia + 1

final = final + avanza

Si final < altura Entonces

final = final -retrocede

FinSi

FinMientras

Mostrar "La rana se demora ", dia, " días en salir"

SiNo

Mostrar "Dato inválido. No puedo calcular la cantidad de días"

FinSi

SiNo

Mostrar "Dato inválido. No puedo calcular la cantidad de días"

FinSi

SiNo

Mostrar "Dato inválido. No puedo calcular la cantidad de días"

FinSi

FinAlgoritmo

**PROBLEMA 2 (20 puntos)**

El siguiente algoritmo calcula el factorial de un número entero:

**Algoritmo Factorial**

```
    Escribir "Ingrese un número y yo encontraré su factorial"
    Leer num
    fact <- 0
    i <- 0

    Mientras (i <= num) hacer
        fact <- fact * i
    FinMientras

    Escribir "el factorial del número ",num, " es ", fact
```

FinAlgoritmo

**PARTE A) (12 puntos)**

El código anterior tiene tres errores. Descúbrelos y:

- Describe cada error encontrado (2 puntos por cada error correctamente identificado;)
- Escribe el pseudo-código corregido (sin los errores lógicos identificados) en la hoja de respuesta (6 puntos)

Error 1: fact debe inicializarse a 1, caso contrario el resultado final sería 0

Error 2: i debe inicializarse a 1, caso contrario el resultado final sería 0

Error 3: ciclo infinito, el i no esta siendo incrementado dentro del while

Pseudocódigo usando mientras

**Algoritmo Factorial**

```
    Escribir "Ingrese un número y yo encontraré su factorial"
    leer num
    fact = 1
    i = 1
    mientras (i<=num)
        fact=fact*i
        i=i+1
    FinMientras
    mostrar 'el factorial del número ',num,' es ',fact
```

FinAlgoritmo

Solución con Para

Algoritmo Factorial

    Escribir "Ingrese un número y yo encontraré su factorial"

    leer num

    fact = 1

    para i=1 hasta num con paso 1

        fact=fact\*i

    FinPara

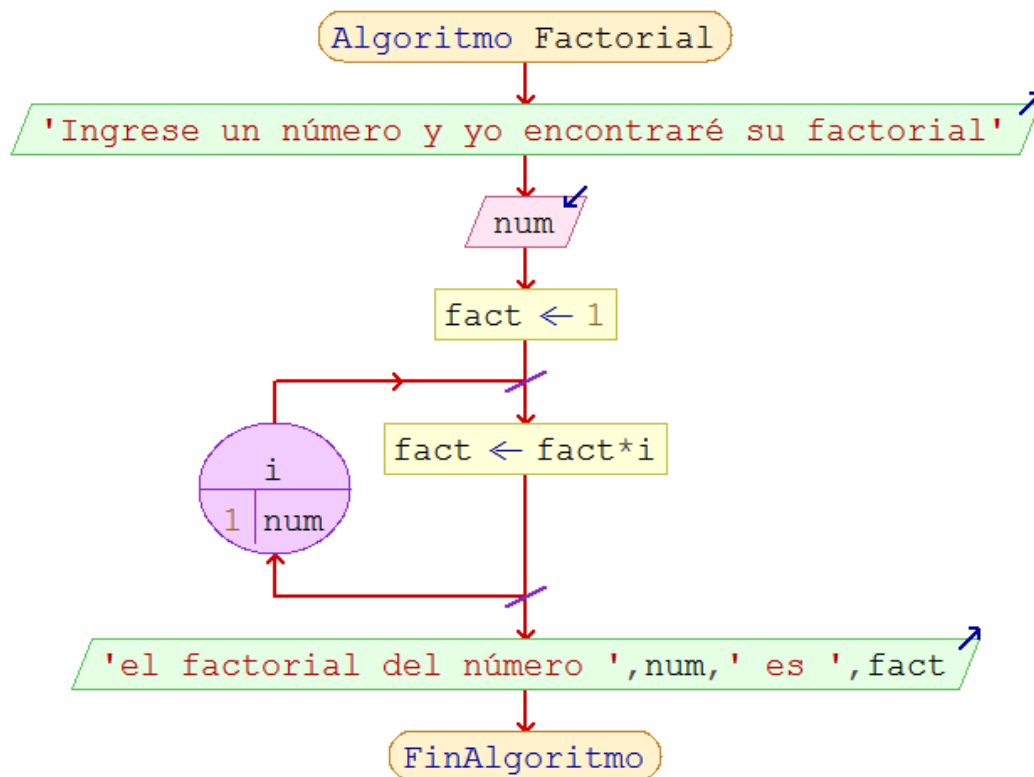
    mostrar 'el factorial del número ',num,' es ',fact

FinAlgoritmo

**PARTE B) (8 puntos)**

Describe el algoritmo corregido de la parte A) **usando un diagrama de flujo.**

Una posible solución



### PROBLEMA 3 (20 puntos)

Dado el diagrama de flujo de la Fig. 3, responde las siguientes preguntas:

#### PARTE A)

Cuando el usuario ingresa el valor 3 ¿qué muestra en pantalla el algoritmo del diagram? **Justifica tu resultado** detallado los cambios de estado {verdadero o falso} a medida que se ejecuta el algoritmo (10 puntos)

Ciclo MIENTRAS: Verdadero

Condicional SI, dentro de ciclo: Falso

Ciclo MIENTRAS: Falso

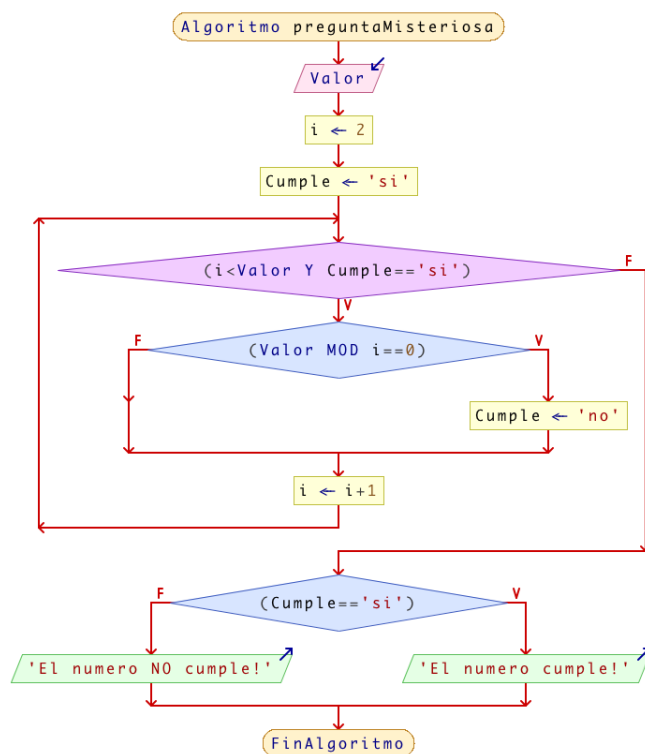
Condicional SI, parte final: Verdadero

Pantalla: "El número cumple"

#### PARTE B)

Escribe el correspondiente algoritmo en pseudo-código (PSeInt). (10 puntos)

Fig. 3. Diagrama de flujo de problema 3



#### Algoritmo preguntaMisteriosa

leer Valor

i=2

Cumple = 'si'

mientras (i<valor y Cumple=='si')

si ((valor%i)==0) Entonces

Cumple = 'no'

FinSi

i=i+1

FinMientras

si (Cumple=='si') Entonces

mostrar 'el numero cumple'

SiNo

mostrar 'el numero no cumple'

FinSi

FinAlgoritmo