

### 1ra Evaluación I Término 2011-2012. Julio 05, 2011

**Tema 1.** (20 puntos). Una persona que deposita **C** dólares en una cuenta de ahorros, el banco le paga una tasa de interés anual **r**, luego de **n** años tendrá un valor acumulado de **A** dólares. La siguiente expresión matemática relaciona estos valores:

$$A = C(1+r)^n$$

Juan y Pedro abren cuentas de ahorros en diferentes bancos.

- En el banco X, Juan deposita en una cuenta de ahorros **C=200** que paga un interés anual de **r=0.08**.
- En el banco Y, Pedro deposita en otra cuenta de ahorros **C=300** que paga un interés anual de **r=0.05**.

Escriba un algoritmo que solicite los datos para las cuentas de Juan y Pedro, determine el año **n** cuando la cantidad **acumulada A de Juan** superará a la cantidad **acumulada A de Pedro**.

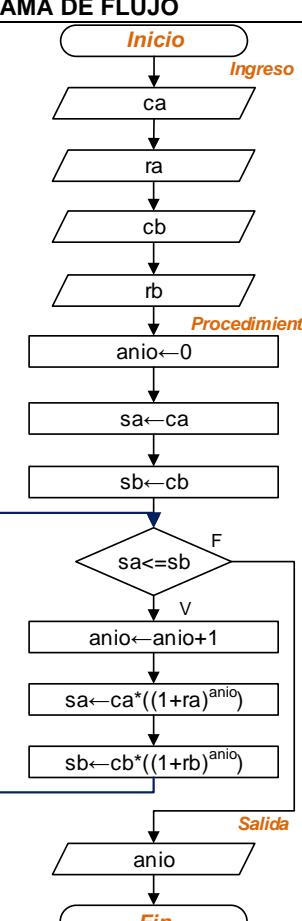
*Nota: Para el algoritmo no se considerarán depósitos o retiros entre los años. Suponga que Juan deposita menos que Pedro y que el interés del Banco X es mayor que Y.*

Rúbrica: Ingreso de datos (5 puntos), determinación de saldos individuales por año (5 puntos), respuesta solicitada (10 puntos)

#### Propuesta de Solución:

Ingresar los datos para cada depositante en variables separadas. Luego, calcular por cada año los saldos para cada depositante, repitiendo hasta que se cumpla la condición que Juan tenga más dinero que Pedro.

La variable “año” es un contador simple que inicia en cero, en finanzas significa el momento en que se deposita. Cuando termina el año completo se considera incrementado en 1.

Descripción	DIAGRAMA DE FLUJO	Python
Inicio		# ICM00794-Fund.Computación - FCNM-ESPOL # 1ra Eval I Término 2011 # Tema 1.Cuentas de Ahorro entre Juan y Pedro # Propuesta: edelros@espol.edu.ec
Ingresar datos de Juan		ca=float(input('Banco A - Capital Inicial : '))
Ingresar datos de Pedro		ra=float(input('Banco A - Tasa Interés Anual:'))
Año inicial		cb=float(input('Banco B - Capital Inicial: '))
Depósito inicial Juan		rb=float(input('Banco B - Tasa Interés Anual:'))
Depósito inicial Pedro		
Mientras el saldo de Juan sea menor o igual que el de Pedro		
Cuenta un año		#Procedimiento
Saldo de Juan		anio=0
Saldo de Pedro		sa=ca
Repita		sb=cb
Mostrar el año		
Fin		while (sa<=sb):
		anio=anio+1
		sa=ca*((1+ra)**anio)
		sb=cb*((1+rb)**anio)
		#Salida
		print(anio)

#### Ejecución del algoritmo: ahorrosjp.py

```
>>>
Banco A - Capital Inicial : 200
Banco A - Tasa Interés Anual:0.08
Banco B - Capital Inicial: 300
Banco B - Tasa Interés Anual:0.05
15
```

```
>>>
Banco A - Capital Inicial : 200
Banco A - Tasa Interés Anual:0.08
Banco B - Capital Inicial: 400
Banco B - Tasa Interés Anual:0.05
25
```

Tarea: Realizar el código en Python usando el diagrama con lazo repita-hasta

Diagrama de Flujo	Python
<pre>     graph TD         Inicio([Inicio]) --&gt; Ingresa[/Ingreso/]         Ingresa --&gt; ca[/ca/]         ca --&gt; ra[/ra/]         ra --&gt; cb[/cb/]         cb --&gt; rb[/rb/]         rb --&gt; Procedimiento[Procedimiento]         Procedimiento --&gt; anio0["anio&lt;-0"]         anio0 --&gt; sa["sa&lt;-ca"]         sa --&gt; sb["sb&lt;-cb"]         sb --&gt; anio1["anio&lt;-anio+1"]         anio1 --&gt; sa["sa&lt;-ca*((1+ra)^anio)"]         sa --&gt; sb["sb&lt;-cb*((1+rb)^anio)"]         sb --&gt; Decision{sa&gt;sb}         Decision -- F --&gt; anio["anio"]         anio --&gt; Fin([Fin])         Decision -- V --&gt; anio1     </pre> <p>The flowchart illustrates a loop structure. It starts with initial values: <math>ca</math>, <math>ra</math>, <math>cb</math>, and <math>rb</math>. It then initializes <math>anio \leftarrow 0</math>. The process continues with <math>sa \leftarrow ca</math> and <math>sb \leftarrow cb</math>. In each iteration, it increments <math>anio \leftarrow anio + 1</math>, calculates <math>sa \leftarrow ca * ((1+ra)^{anio})</math>, and calculates <math>sb \leftarrow cb * ((1+rb)^{anio})</math>. A decision diamond checks if <math>sa &gt; sb</math>. If false (F), it outputs <math>anio</math> and ends at <b>Fin</b>. If true (V), it loops back to increment <math>anio</math> again.</p>	