**Algoritmo para saber cuántos billetes de cada denominación debe entregar un cajero en la ciudad de Valledupar**

**Nombre: Iván Andrés Utria Álvarez**

**Docente: Juan Janet**

**Universidad Popular del Cesar**

**Valledupar**

**2022 – 2**

**Problemática**

Se busca saber cuantos billetes de cada denominación debe entregar un cajero automático cuando un usuario quiera retirar una cantidad X de dinero. Para ello se ha encargado realizar un algoritmo que simule un retiro y muestre en pantalla la cantidad de dinero a entregar, en billetes de **$100.000**, **$50.000**, **$20.000** y **$10.000**.

**Lógica del algoritmo**

**Paso 1:** Definir cuantos billetes de cada denominación hay en el cajero, para ello se declaran 4 variables para definir la cantidad de billetes

1. num\_cien
2. num\_cincuenta
3. num\_veinte
4. num\_diez

**paso 2: conocer el valor que se desea retirar**

valor\_retiro

**Paso 3: usar una variable auxiliar, esta variable va a comenzar con el valor del retiro e ir decrementando su valor a medida que se van calculando el numero de billetes a entregar, si la cantidad de billetes a entregar es mayor a cero**

AUX = valor\_retiro

**Paso 4: saber cuántos billetes de $100.000 se debe entregar, N es una variable que contendrá el numero de billetes a entregar**

N = parte\_entera(AUX / 100000)

**Pasos 5: verificar si se puede entregar esa cantidad de billetes**

Si **num\_cien = 0** entonces

Cantidad\_cien\_entregar = 0

sino

si **num\_cien < N** entonces

Cantidad\_cien\_entregar = num\_cien

Sino

Cantidad\_cien\_entregar = N

Fin si

Fin si

**Paso 6: se modifica el valor de AUX**

AUX = AUX – (Cantidad\_cien\_entregar \* 100000)

**Paso 5: se repiten los pasos del 4 al 6 con los valores de $50.000, $20.000, $10.000**

**Para $50.000**

N = parte\_entera(AUX / 50000)

Si **num\_cincuenta = 0** entonces

Cantidad\_cincuenta\_entregar = 0

sino

si **num\_cincuenta < N** entonces

Cantidad\_cincuenta\_entregar = num\_cincuenta

Sino

Cantidad\_cincuenta\_entregar = N

Fin si

Fin si

AUX = AUX – (Cantidad\_cincuenta\_entregar \* 50000)

**Para $20.000**

N = parte\_entera(AUX / 20000)

Si **num\_**veinte **= 0** entonces

Cantidad\_veinte\_entregar = 0

sino

si **num\_**veinte **< N** entonces

Cantidad\_veinte\_entregar = num\_veinte

Sino

Cantidad\_veinte\_entregar = N

Fin si

Fin si

AUX = AUX – (Cantidad\_veinte\_entregar \* 20000)

**Para $10.000**

N = parte\_entera(AUX / 10000)

Si **num\_**diez **= 0** entonces

Cantidad\_diez\_entregar = 0

sino

si **num\_**diez **< N** entonces

Cantidad\_diez\_entregar = num\_ diez

Sino

Cantidad\_diez\_entregar = N

Fin si

Fin si

AUX = AUX – (Cantidad\_diez\_entregar \* 10000)

**Paso 7: se descuenta el número de billetes de cada denominación, dependiendo del numero de billetes a entregar**

num\_cien = num\_cien - Cantidad\_cien\_entregar

num\_cincuenta = num\_cincuenta - Cantidad\_cincuenta\_entregar

num\_veinte = num\_veinte - Cantidad\_veinte\_entregar

num\_diez = num\_diez - Cantidad\_diez\_entregar

**Como se puede notar, este algoritmo es repetitivo y se puede mejorar. ¿Cómo…?**

1. Para ello se debe declarar un array bidimensional que contenga las cantidades y denominaciones de los billetes.
2. También declarar un vector en el cual se va a guardar el numero de billetes que se va a entregar.

**Algoritmo final en seudocódigo**

**Algoritmo**

**//definir un array de dos dimensiones con la cantidad de billetes de cada denominacion,**

**//indicando en la pos 1 la denominacion y en la pos 2 la cantidad de billetes**

**Dimension billetes[4,2];**

**billetes[1,1] <- 100000;**

**billetes[1,2] <- 17;**

**billetes[2,1] <- 50000;**

**billetes[2,2] <- 28;**

**billetes[3,1] <- 20000;**

**billetes[3,2] <- 15;**

**billetes[4,1] <- 10000;**

**billetes[4,2] <- 64;**

**//definir array para guardar la cantidad de billetes a entregar**

**Dimension Retiro[4];**

**//definir variable para la cantidad de dinero que se va a retirar**

**Definir cantidad\_retirar, num\_billetes Como Entero;**

**//pedir la cantidad de dinero a retirar**

**Escribir "ingrese la cantidad de dinero a retirar";**

**leer cantidad\_retirar;**

**//validar la cantidad a retirar**

**si(cantidad\_retirar <= 0) Entonces**

**Escribir "cantidad no valida";**

**SiNo**

**AUX <- cantidad\_retirar;**

**Para i<-1 Hasta 4 Con Paso 1 Hacer**

**num\_billetes <- trunc(AUX / billetes[i,1]);**

**Si (billetes[i,2]=0) Entonces**

**Retiro[i] <- 0;**

**SiNo**

**Si (billetes[i,2] < num\_billetes) Entonces**

**Retiro[i] <- billetes[i,2];**

**SiNo**

**Retiro[i] <- num\_billetes;**

**Fin Si**

**Fin Si**

**AUX <- AUX - (Retiro[i] \* billetes[i,1]);**

**Fin Para**

**FinSi**

**Para i <- 1 Hasta 4 Con Paso 1 Hacer**

**billetes[i,2] <- billetes[i,2] - Retiro[i]**

**Fin Para**

**Para i <- 1 Hasta 4 Con Paso 1 Hacer**

**Escribir "se deben entregar ", Retiro[i], " billetes de ", billetes[i,1]**

**Fin Para**

**FinAlgoritmo**

**Observaciones:** debe tener en cuenta una serie de validaciones como son:

1. Validar que la cantidad a retirar sea múltiplo de 10000
2. Validar que la cantidad de billetes a entregar corresponda a la cantidad de dinero que solicito retirar
3. Agregar un limite para la cantidad de billetes de $100.000 que se pueden entregar por retiro

Para ver este algoritmo en la versión web ingrese a <https://github.com/iutria/cajero>