

UNIVERSIDAD NACIONAL DEL SANTA
FACULTAD DE INGENIERÍA
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMÁTICA



TEMA

Algoritmo en Java y Pseint

ESTUDIANTE

Angeles Rodriguez Ilan Nestor

PROFESOR

Escalante Espinoza Walter Jesús

CURSO

Algoritmos

Nuevo Chimbote

Perú

2022

EJERCICIO DE ALGORITMO

Ejercicio: Ejemplo con 4 subprocesos con argumentos por valor y referencia. Poner enunciado. Desarrollado en pseint y java. Presentarlo en Word hasta el día domingo 21 de agosto. 13 horas.

Resumen: La finalidad de este informe es realizar un algoritmo que registre el proceso de una billetera digital. Para tal fin se hará uso de ciclos y arrays, temas aprendidos en el curso de algoritmos.

Palabras claves: Token, java, pseudocódigo, algoritmo, funciones, subpproceso.

Abstract: The purpose of this report is to perform an algorithm that records the process of a digital wallet. For this purpose, cycles and arrays will be used, topics learned in the course of algorithms.

Keywords: Token, java, pseudocode, algorithm, functions, thread.

1. Algoritmo en Pseint

//SubAlgoritmo para el inicio de sesion

SubAlgoritmo login(id Por Referencia)

Escribir "-----"

Escribir "|=== Red principal de Ethereum ===|"

Escribir "-----"

Escribir "=== Bienvenido a la EPISIWallet ==="

Escribir "-----"

Escribir "Introduce tu direccion:" sin saltar

leer id

Escribir "Introduce tu clave:" Sin Saltar

leer password

FinSubAlgoritmo

SubProceso iniciarSesion

wait

Escribir "Iniciando Sesion....."

wait

FinSubProceso

//Agrego tiempo de espera

SubProceso wait

esperar 2 Segundos

borrar pantalla

FinSubProceso

//Opciones de la billetera

SubProceso menu(id Por Referencia, episicoin Por Referencia , monto Por Referencia,
ns Por Referencia, compra Por Referencia)

episicoin=episicoin-monto+compra/100

ns=episicoin*100

Escribir " ID:" id

Escribir "|===== " episicoin " EPISICoin =====|"

```
Escribir "----- S/. " ns "-----"
escribir " " "1.Enviar" " " Sin Saltar
escribir "2.Comprar" " " Sin Saltar
escribir "3.Activos" " "
escribir "4.Actividad" " " Sin Saltar
escribir "5.Otro" " " Sin Saltar
escribir "6.Cerrar Sesion"
escribir "Seleccionar:" Sin Saltar
leer opc
segun opc hacer
    1: enviar(id, episicoin, monto)
    2: Escribir "Comprar"
        comprar(id, episicoin, monto, ns)
    3: Escribir "Activos"
        agregarActivo
        //escribir token[a]
    4: Escribir "Actividad"
    5: outro(id)
    6: cerrarSesion
FinSegun
wait
FinSubProceso
//Opciones complementarias
SubProceso enviar(id Por Referencia, episicoin Por Referencia, monto Por Referencia)
    wait
    escribir "----- Enviar a -----"
    escribir "introduce la direccion:" Sin Saltar
    Leer id2
    wait
    Escribir "Buscando...."
```

```
wait
escribir "Activo:" episicoin " EPISicoins"
escribir "Importe:" Sin Saltar
leer monto
escribir monto " EPISicoins"
escribir "Confirmar:"
escribir "1. Si"
escribir "2. No"
leer z
si z=1 Entonces
    Escribir "Envio exitoso"
SiNo
    enviar(id, episicoin, monto)
FinSi
wait
menu(id, episicoin ,monto, ns, compra)
```

FinSubProceso

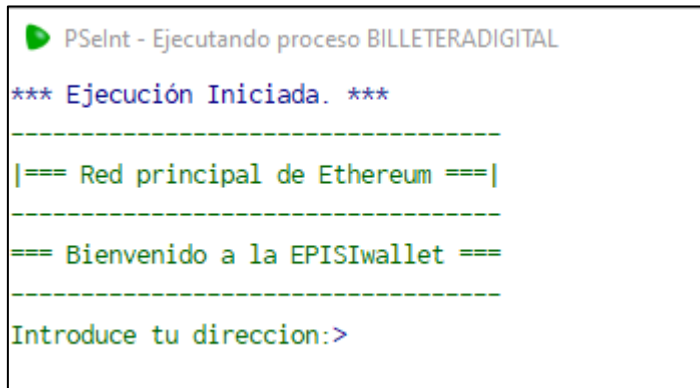
SubProceso comprar(id Por Referencia, episicoin Por Referencia, monto Por Referencia, ns Por Referencia)

```
wait
Escribir "----- Comprar EPISicoins -----"
Escribir "Activo: S/." ns
Escribir "Importe:" sin saltar
leer compra
escribir "S/." compra " == " compra/100 " EPISicoins"
escribir " "
escribir "Confirmar:"
escribir "1. Si"
escribir "2. No"
leer z
si z=1 Entonces
```

```
        Escribir "Compra exitosa"
    SiNo
        enviar(id, episicoin, monto)
    FinSi
    wait
    menu(id, episicoin, monto, ns, compra)
FinSubProceso
SubProceso outro(id Por Referencia)
    wait
    escribir "|===== ¿No ves tu token? =====|"
    Escribir "1.Actualizar lista" " " " Sin Saltar
    Escribir "2.Agregar activo"
    leer opc2
    segun opc2 Hacer
        1: wait
            //menu
        2: escribir "Agregar activo"
            agregarActivo
            escribir "===== Activo agregado ====="
            wait
            //menu
    FinSegun
    menu(id, episicoin, monto, ns, compra)
FinSubProceso
SubProceso agregarActivo
    wait
    dimension token[3]
    token[0]<-"BTC" //Bitcoin
    token[1]<-"ETH" //Ethereum
    token[2]<-"SOL" //Solana
```

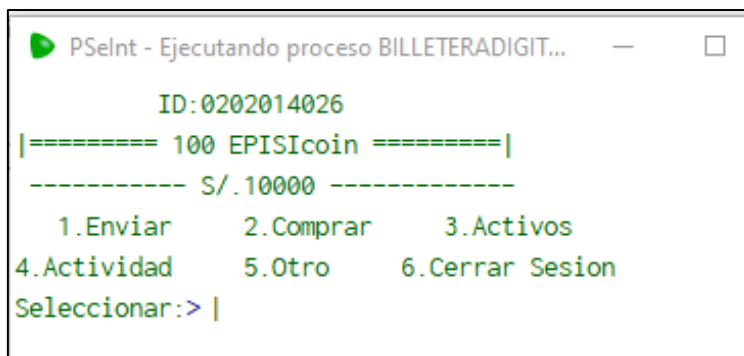
```
Para i<-0 hasta 2 con paso 1 Hacer
    x=x+1
    Escribir x "." token[i]
FinPara
Para i<-0 hasta 0 con paso 1 Hacer
    Escribir "Selecciona activo:" Sin Saltar
    leer a
FinPara
Para i<-0 hasta 0 con paso 1 Hacer
    Escribir "Agregaste el activo:" token[a-1]
FinPara
wait
menu(id, episicoin ,monto, ns, compra)
FinSubProceso
//Cerrar
SubProceso cerrarSesion
    wait
    Escribir "Cerrando Sesion....."
    wait
    escribir "Gracias"
FinSubProceso
Algoritmo billeteraDigital
    //Bilitera digital para el token EPISicoin
    episicoin=100
    //episicoin=100*ns
    login(id)
    iniciarSesion
    menu(id, episicoin,monto, ns, compra)
FinAlgoritmo
```


2. Ejecución en pseint



```
PSeInt - Ejecutando proceso BILLETERADIGITAL
*** Ejecución Iniciada. ***
-----
|=== Red principal de Ethereum ===|
-----
=== Bienvenido a la EPISiwallet ===
-----
Introduce tu direccion:>
```

Ilustración 1: Inicio de la billetera




```
PSeInt - Ejecutando proceso BILLETERADIGIT...
ID:0202014026
|===== 100 EPISiCoin =====|
----- S/.10000 -----
1.Enviar      2.Comprar      3.Activos
4.Actividad   5.Otro        6.Cerrar Sesión
Seleccionar:> |
```

Ilustración 3: Inicio de la billetera

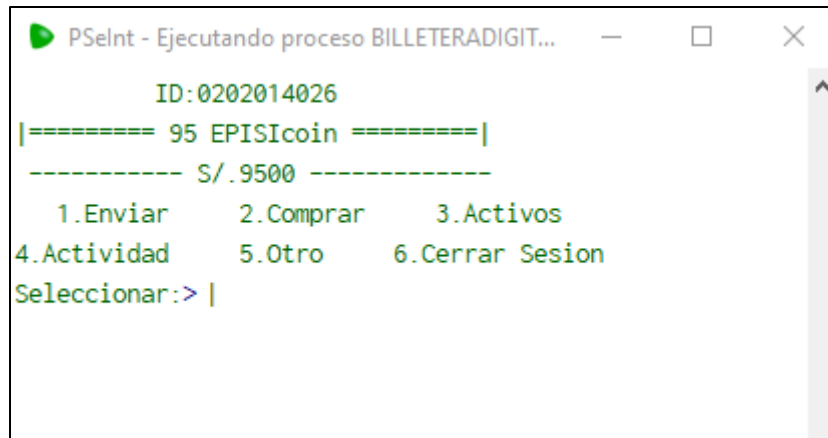


```
PSeInt - Ejecutando proceso BILLETERADIGIT...
----- Enviar a -----
introduce la direccion:> 0202114026
```

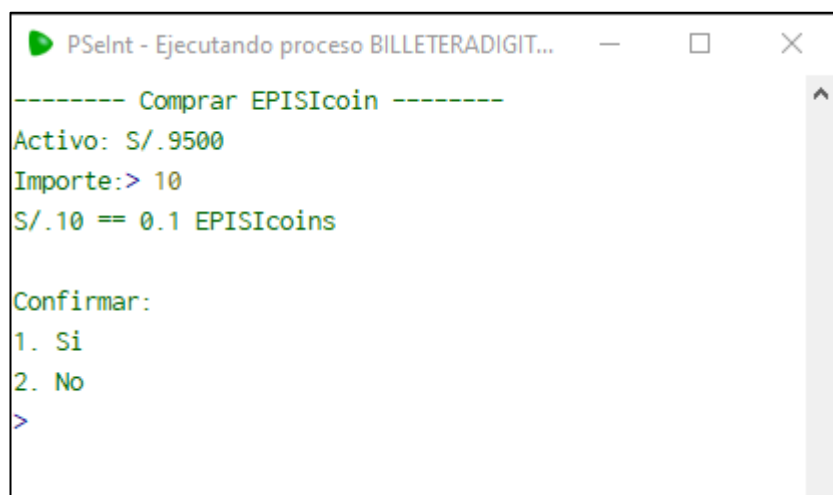
Ilustración 2: Dirección de cuenta



```
PSeInt - Ejecutando proceso BILLETERADIGIT...
Activo:100 EPISiCoins
Importe:> 5
5 EPISiCoins
Confirmar:
1. Si
2. No
>
```



```
PSeInt - Ejecutando proceso BILLETARADIGIT...  
  
ID:0202014026  
|===== 95 EPISICoin =====|  
----- S/.9500 -----  
1.Enviar      2.Comprar      3.Activos  
4.Actividad   5.Otro        6.Cerrar Sesion  
Seleccionar:> |
```

Ilustración 5: Actualizacion de cuenta

```
PSeInt - Ejecutando proceso BILLETARADIGIT...  
  
----- Comprar EPISICoin -----  
Activo: S/.9500  
Importe:> 10  
S/.10 == 0.1 EPISICoins  
  
Confirmar:  
1. Si  
2. No  
>
```

Ilustración 4: Compra

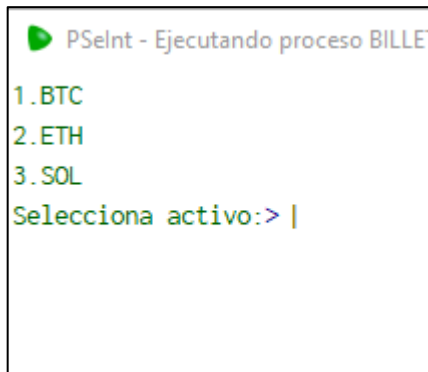
```
PSeInt - Ejecutando proceso BILLETARADIGIT...  
  
ID:0202014026  
|===== 90.1 EPISICoin =====|  
----- S/.9010 -----  
1.Enviar      2.Comprar      3.Activos  
4.Actividad   5.Otro        6.Cerrar Sesion  
Seleccionar:>
```



```

PSeInt - Ejecutando proceso BILLETADIGIT...
|===== ¿No ves tu token? =====|
1.Actualizar lista  2.Agregar activo
>

```



```

PSeInt - Ejecutando proceso BILLE
1.BTC
2.ETH
3.SOL
Selecciona activo:> |

```

3. Algoritmo en Java

```

/*
 * @IlanNestorAngelesRodriguez
 */
import java.util.Scanner;

public class billeteraDigital {
    public static void main(String[] args) {
        int id, password;
        double episicoin;
        episicoin=100;
        System.out.println("-----");
        System.out.println("|=== Red principal de Ethereum ===|");
        System.out.println("-----");
        System.out.println("=== Bienvenido a la EPISIWallet ===");
        Scanner teclado = new Scanner(System.in);
        System.out.print("Introduce tu direccion:");
        id=teclado.nextInt();
        System.out.print("Introduce tu clave:");

```

```

        password=teclado.nextInt();

        wait(2);

        iniciarsesion();

        wait(2);

        menu();

    }

    public static void iniciarsesion(){

        System.out.println("Iniciando Sesion.....");

    }

    public static void wait(int segundos){

        try {

            Thread.sleep(segundos * 1000);

        } catch (Exception e) {

            System.out.println(e);

        }

    }

    public static void menu(){

        int opc, id2, monto, z, compra, i, x=0, a=1, opc2;

        /*episicoin=episicoin-monto+compra/100;*/

        /*ns=episicoin*100;*/

        /*System.out.println("      ID:" + id);

        System.out.println("|===== " + episicoin + " EPISicoin
        =====|");

        System.out.println(" ----- S/." + ns + " -----");*/

        System.out.print(" " + "1.Enviar" + " ");

        System.out.print("2.Comprar" + " ");

        System.out.println("3.Activos" + " ");

        System.out.print("4.Actividad" + " ");

        System.out.print("5.Otro" + " ");

        System.out.println("6.Cerrar Sesion");

        System.out.println("Seleccionar:");

```

```
Scanner teclado = new Scanner(System.in);

opc=teclado.nextInt();
switch (opc) {
case 1:
    System.out.println("----- Enviar a -----");
    System.out.print("introduce la direccion:");
    id2=teclado.nextInt();
    /*wait(2)*/
    System.out.println("Buscando....");
    /*wait(2)*/
    /*System.out.println("Activo:" + episicoin + " EPISicoins");*/
    System.out.println("Importe:");
    monto=teclado.nextInt();
    System.out.println(monto + " EPISicoins");
    System.out.println("Confirmar:");
    System.out.println("1. Si");
    System.out.println("2. No");
    z=teclado.nextInt();
    if(z==1){
        System.out.println("Envio exitoso");
    }
    if(z==0){
        System.out.println("Accion cancelada");
        /*enviar(id, episicoin, monto)*/
    }
    /*menu(id, episicoin ,monto, ns, compra)*/
    break;
case 2:
    System.out.println("----- Comprar EPISicoin -----");
    /*System.out.println("Activo: S/." + ns);*/
```

```
System.out.print("Importe:");
compra=teclado.nextInt();
System.out.println("S/." + compra + " == " + compra/100 + "
EPISICOINS");
System.out.println(" ");
System.out.println("Confirmar");
System.out.println("1. Si");
System.out.println("2. No");
z=teclado.nextInt();
if(z==1){
    System.out.println("Compra exitoso");
}
if(z==0){
    System.out.println("Accion cancelada");
    /*enviar(id, episicoin, monto)*/
}
/*menu(id, episicoin, monto, ns, compra)*/
break;
case 3:
    String [] token= new String[3];
    token[0]="BTC";
    token[1]="ETH";
    token[2]="SOL";

    for (i=0;i<=2;i++) {
        x=x+1;
        System.out.println(x+"."+token[i]);
    }
    for (i=0;i<=0;i++) {
        System.out.print("Selecciona activo:");
        a=teclado.nextInt();
```

```
        }
        for (i=0;i<=0;i++) {
            System.out.println("Agregaste el activo:" + token[a-1]);
        }
        break;
case 4:
    System.out.println("En proceso.....");
    break;
case 5:
    System.out.println("|===== ¿No ves tu token? =====|");
    System.out.print("1.Actualizar lista" + " ");
    System.out.println("2.Agregar activo");
    opc2=teclado.nextInt();
    switch (opc2) {
        case 1: /*wait*/

            break;
        case 2: System.out.println("Agregar activo");
            //case 3
            System.out.println("===== Activo agregado =====");
        default:
            throw new AssertionError();
    }

    //menu
menu(id, episicoin, monto, ns, compra)*/
    break;
default:
    System.out.println("Selecciona la opcion correcta");;
} while(opc != 0);
System.out.println("Se termino el programa.....");
```

```

    }
}

```

4. Ejecución del algoritmo en la terminal de comandos

```

C:\Users\hp\Downloads\Billetera_Digital>java billeteraDigital
Error: Could not find or load main class billeteraDigital
Caused by: java.lang.ClassNotFoundException: billeteraDigital

C:\Users\hp\Downloads\Billetera_Digital>javac billeteraDigital.java

C:\Users\hp\Downloads\Billetera_Digital>java billeteraDigital
=====
|=== Red principal de Ethereum ===|
=====
=== Bienvenido a la EPISIWallet ===
Introduce tu direccion:0202014026
Introduce tu clave:1214
Iniciando Sesion.....
    1.Enviar      2.Comprar      3.Activos
    4.Actividad   5.Otro        6.Cerrar Sesion
Seleccionar:

```

Ilustración 6: Compilación y ejecución

```

|=== Red principal de Ethereum ===|
=====
=== Bienvenido a la EPISIWallet ===
Introduce tu direccion:0202014026
Introduce tu clave:1214
Iniciando Sesion.....
    1.Enviar      2.Comprar      3.Activos
    4.Actividad   5.Otro        6.Cerrar Sesion
Seleccionar:
2
----- Comprar EPISICoin -----
Importe:12
S/.12 == 0 EPISICoins

Confirmar
1. Si
2. No
2

```



```
=====
|=== Red principal de Ethereum ===|
=====
=== Bienvenido a la EPISIWallet ===
Introduce tu direccion:0202014026
Introduce tu clave:12
Iniciando Sesion.....
    1.Enviar      2.Comprar      3.Activos
4.Actividad      5.Otro          6.Cerrar Sesion
Seleccionar:
1
----- Enviar a -----
introduce la direccion:125468
Buscando....
Importe:
12
12 EPISICOINS
Confirmar:
1. Si
2. No
1
Envio exitoso
```

Ilustración 7: Ejecución en la terminal de comandos

Tabla de Ilustraciones

Ilustración 1:Inicio de la billetera..... 8

Ilustración 3: Inicio de la billetera..... 8

Ilustración 2: Dirección de cuenta 8

Ilustración 4: Compra 9

Ilustración 5: Actualizacion de cuenta..... 9

Ilustración 6: Compilación y ejecución 9

Ilustración 7: Ejecución en la terminal de comandos 9