

# Computación I - Práctica II

Navarro Miranda, Mauricio  
mauricio@navarromiranda.mx

Arévalo Loyola, Alma Rosario  
olaverax@gmail.com

25 de agosto de 2014

## Resumen

Comprender el uso de variables y utilizarlas junto con instrucciones *if* para tomar decisiones.

## 1. Realiza las siguientes conversiones:

a)  $1000_8$  a binario.

En el sistema octal y en el binario, cada dígito indica el número de veces que se suma cada potencia de 8 o de 2, respectivamente, según el orden.

$$\begin{aligned} 1000_8 &= 1(8^3) + 0(8^2) + 0(8^1) + 0(8^0) \\ &= 1(2^9) + 0(2^8) + 0(2^7) + 0(2^6) + 0(2^5) + 0(2^4) + 0(2^3) + 0(2^2) + 0(2^1) + 0(2^0) \\ &= 1000000000_2 \end{aligned}$$

b)  $101110101_2$  a hexadecimal.

Como  $16 = 2^4$  podemos agrupar de 4 en 4 dígitos, y cada uno de estos grupos representa un número menor que 16, por lo tanto se puede escribir como un dígito en sistema hexadecimal.

$$1/0111/0101$$

$$1_2 = 1, 0111_2 = 7, 0101_2 = 5$$

De modo que

$$101110101_2 = 175_{16}$$

c)  $100101100_2$  a decimal.

Nuevamente, sumando las potencias de 2, según el orden de los dígitos.

$$\begin{aligned} 100101100_2 &= 1(2^8) + 0(2^7) + 0(2^6) + 1(2^5) + 0(2^4) + 1(2^3) + 1(2^2) + 0(2^1) + 0(2^0) \\ &= 256 + 32 + 8 + 4 = 300_{10} \end{aligned}$$

d)  $125_{10}$  a binario.

125 expresado como suma de potencias de 2 es:

$$\begin{aligned} 125 &= 64 + 32 + 16 + 8 + 4 + 1 \\ &= 1(2^6) + 1(2^5) + 1(2^4) + 1(2^3) + 1(2^2) + 0(2^1) + 1(2^0) \\ &= 1111101_2 \end{aligned}$$

## 2. Indica qué número entero representan los siguientes números binarios, considerando a) entero no signado y b) entero signado:

Considerando como entero no signado es el mismo procedimiento que en el inciso c) del ejercicio anterior:

**a.a)**  $1100_2 = 1(2^3) + 1(2^2) + 0(2^1) + 0(2^0) = 8 + 4 = 12$

**a.b)**  $1101_2 = 1(2^3) + 1(2^2) + 0(2^1) + 1(2^0) = 8 + 4 + 1 = 13$

**a.c)**  $0011_2 = 0(2^3) + 0(2^2) + 1(2^1) + 1(2^0) = 2 + 1 = 3$

Considerando como entero signado, el primer dígito representa el signo ('0' significa positivo y '1' negativo).

**b.a)** 1100 Quitamos el primer 1, que significa que el número es negativo, y nos queda 100, obtenemos su complemento a dos, que es 100, y convertido a decimal es

$$100_2 = 1(2^2) + 0(2^1) + 0(2^0) = 4$$

Por lo tanto, éste número representa al  $-4$

**b.b)** 1101 Quitamos el primer 1, que significa que el número es negativo, y nos queda 101, obtenemos su complemento a dos, que es 011, y convertido a decimal es

$$11_2 = 1(2^1) + 1(2^0) = 2 + 1 = 3$$

Por lo tanto, éste número representa al  $-3$

**b.c)** 0011 Como su primer dígito es 0, este número es positivo y se convierte de la manera convencional.

$$11_2 = 1(2^1) + 1(2^0) = 2 + 1 = 3$$

Los códigos de las aplicaciones 3 a 5 están en las siguientes páginas (un código por página).

También están incluidos en el directorio src

Y pueden ser consultados en github:

[https://github.com/mautematico/java.ciencias/tree/master/P\\_0011](https://github.com/mautematico/java.ciencias/tree/master/P_0011)

### 3. Escribe una aplicación que lea cinco enteros y que determine e imprima los enteros mayor y menor en el grupo.

src/MinimoYmaximo.java

```
1  /*
2  v:0.1.1
3  Programa que lee cinco enteros,
4  determina el mínimo y el máximo e imprime a pantalla mínimo y máximo.
5  Mauricio Navarro Miranda, Alma Arévalo Loyola
6  Facultad de Ciencias
7  */
8  //Importamos el paquete necesario para usar Scanner
9  import java.util.Scanner;
10 public class MinimoYmaximo{
11     public static void main(String args[]){
12         //Preparamos una instancia de Scanner para leer de la entrada estándar
13         Scanner entrada = new Scanner(System.in);
14
15         //Pedimos al usuario el primer entero; lo guardamos en actual. Notemos que éste es,
16         //hasta ahora, el máximo el mínimo.
17         System.out.print("Dame el primer entero... ");
18         int actual = entrada.nextInt();
19         int maximo = actual;
20         int minimo = actual;
21
22         System.out.print("Dame el segundo entero... ");
23         //leemos el segundo entero
24         actual = entrada.nextInt();
25         //Si el entero que acabamos de leer es mayor que nuestro mayor almacenado,
26         //actualizamos mayor.
27         if(actual > maximo)
28             maximo = actual;
29         //Análogamente para el menor.
30         if(actual < minimo)
31             minimo = actual;
32
33         //Repetimos tres veces más. Omitiré los comentarios a propósito.
34         System.out.print("Dame el tercer entero... ");
35         actual = entrada.nextInt();
36         if(actual > maximo)
37             maximo = actual;
38         if(actual < minimo)
39             minimo = actual;
40
41         System.out.print("Dame el cuarto entero... ");
42         actual = entrada.nextInt();
43         if(actual > maximo)
44             maximo = actual;
45         if(actual < minimo)
46             minimo = actual;
47
48         System.out.print("Dame el quinto entero... ");
49         actual = entrada.nextInt();
50         if(actual > maximo)
51             maximo = actual;
52         if(actual < minimo)
53             minimo = actual;
54
55         //Ya hemos leído los cinco enteros. Ahora, int maximo tiene al máximo de ellos,
56         //mientras que el mínimo está en int minimo.
57         System.out.printf("El máximo de los enteros es %d, y el mínimo es %d\n", maximo,
58                             minimo);
59     }
60 }
```

4. Escribe una aplicación que lea un entero y que determine e imprima si es impar o par.

src/ParOno.java

```
1  /*
2      v:0.1
3      Programa que lee un entero,      determina e imprime si es par o impar.
4      Mauricio Navarro Miranda, Alma Arévalo Loyola
5      Facultad de Ciencias.
6  */
7  //Importamos el paquete necesario para usar Scanner
8  import java.util.Scanner;
9  public class ParOno{
10     public static void main(String args[]){
11         //Preparamos una entrada de Scanner, para leer la entrada estándar.
12         Scanner entrada = new Scanner(System.in);
13
14         //Pedimos al usuario que nos proporcione el entero que vamos a evaluar.
15         System.out.print("Dame un número entero... ");
16         //Leemos el siguiente entero en la entrada
17         int numero = entrada.nextInt();
18
19         //Si el número es congruente a 1 (módulo 2), entonces es impar.
20         if(numero % 2 == 1)
21             System.out.printf("%d es un número impar\n", numero);
22         //Si el número es congruente a 0 (módulo 2), entonces es par.
23         if(numero % 2 == 0)
24             System.out.printf("%d es un número par\n", numero);
25     }
26 }
27 }
```

5. Escribe una aplicación que calcule los cuadrados y cubos de los números del 0 al 10, y que imprima los valores resultantes en formato de tabla.

src/Potencias.java

```
1  /*
2  v:0.1
3  Programa que calcula e imprime los cuadrados y los cubos de los primeros once naturales.
4  Mauricio Navarro Miranda, Alma Arévalo Loyola
5  Facultad de Ciencias.
6  */
7  public class Potencias{
8      public static void main(String args[]){
9          int numero = 0;
10
11          System.out.println("Número\tCuadrado\tCubo");
12          //Iniciamos con numero = 0
13          System.out.printf(" %d\t%d\t\t%d\n", numero, numero*numero, numero*numero*numero);
14
15          /*
16             Ahora, haremos diez veces lo siguiente:
17             incrementaremos el número en 1.
18             imprimiremos la fila de la tabla con el nuevo número, su cuadrado y su cubo.
19          */
20          numero+=1;
21          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
22          numero+=1;
23          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
24          numero+=1;
25          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
26          numero+=1;
27          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
28          numero+=1;
29          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
30          numero+=1;
31          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
32          numero+=1;
33          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
34          numero+=1;
35          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
36          numero+=1;
37          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
38          numero+=1;
39          System.out.printf(" %d\t%d\t\t\t%d\n", numero, numero*numero, numero*numero*numero);
40      }
41  }
```