

Ejercicio 2

A.

char : Entra a int: 103

- char c = 'g'; se promociona a int automáticamente porque char se considera un tipo entero en Java (valor Unicode de 'g' es 103).

short : Entra a int: 2

- No hay una sobrecarga directa para short, por lo que short se promueve a int.

byte : Entra a int: 1

- no hay una sobrecarga directa para byte, por lo que se promueve a int.

long : Entra a double: 9.99999999E8

- Se promueve a double ya que no puede ser promovido a int al no caber un long necesariamente en un int

integer : Entra a int: 51232

- Int coincide con el método función(int a)

double : Entra a double: 12.4

- double coincide directamente con la sobrecarga de función(double a).

float : Entra a double: 5.650000095367432

- Al no haber sobrecarga directa para float, entonces se promueve a double

B.

Short

Al activar el método que recibe un short el único cambio es que ahora

short : Entra a short: 2 ya que ya existe un método para short.

Float

Ahora long : Entra a float: 1.0E9 ya que al ahora existir un método float, el long pasa primero por float y no debe irse hasta el double, lo mismo pasa con

float : Entra a float: 5.65

//Double y Float

Se produce un error de compilación ya que no hay un método sobrecargado para double, y double no puede promocionarse ni a float ni a int

//Todas menos double

Ahora todos los métodos pasan a double debido a la conversión implícita de los tipos numéricos más pequeños