Mercado Salinas Fhernando

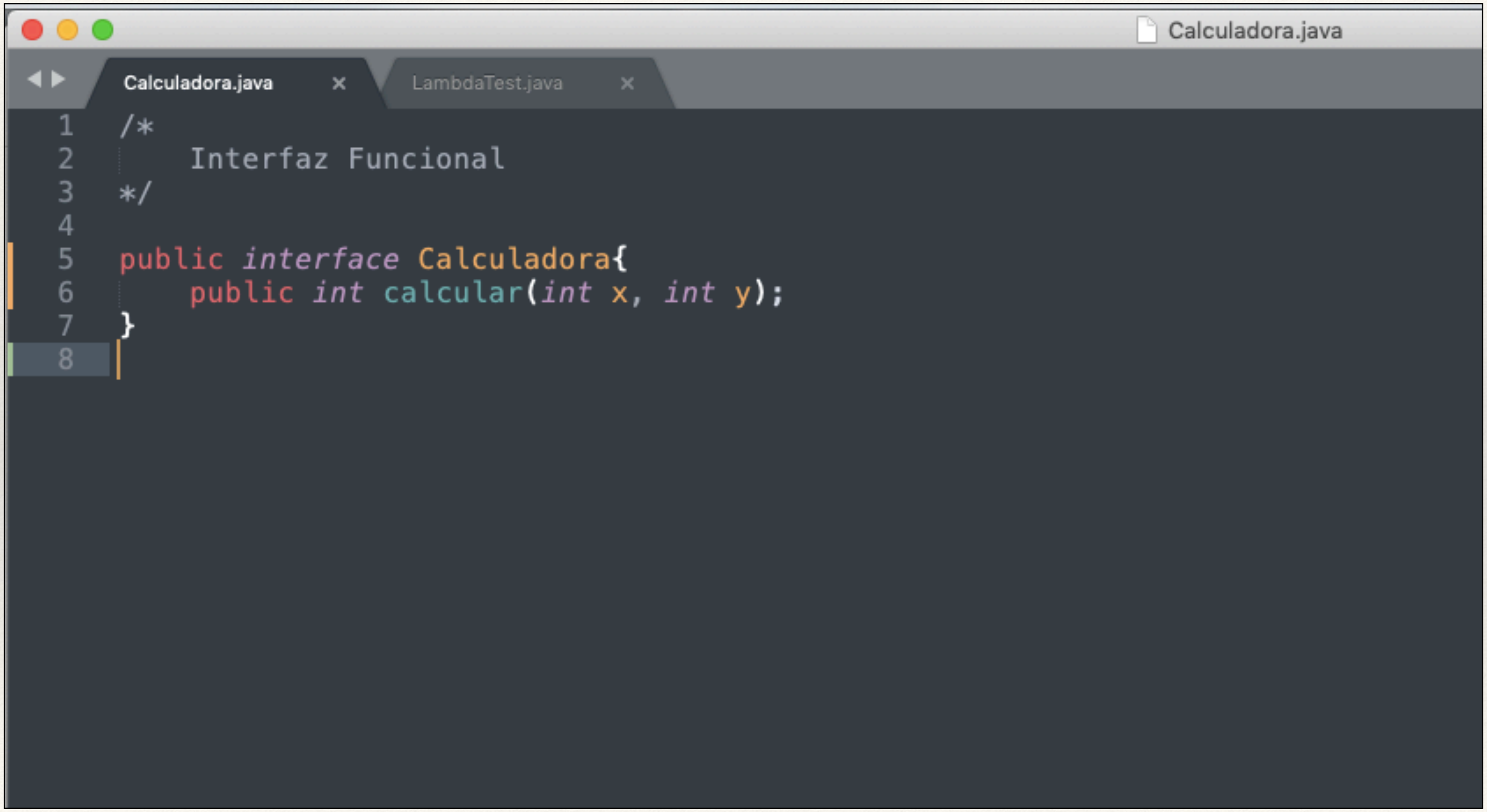
- *Máster Universitario en Sistemas Embebidos*
  - Mondragon Unibertsitatea, Basque Country, Spain

- *Ingeniería en Sistemas Computacionales*
  - Tecnológico de Estudios Superiores de Jocotitlán

Programación Lógica y Funcional

v 0.0, Advanced Level

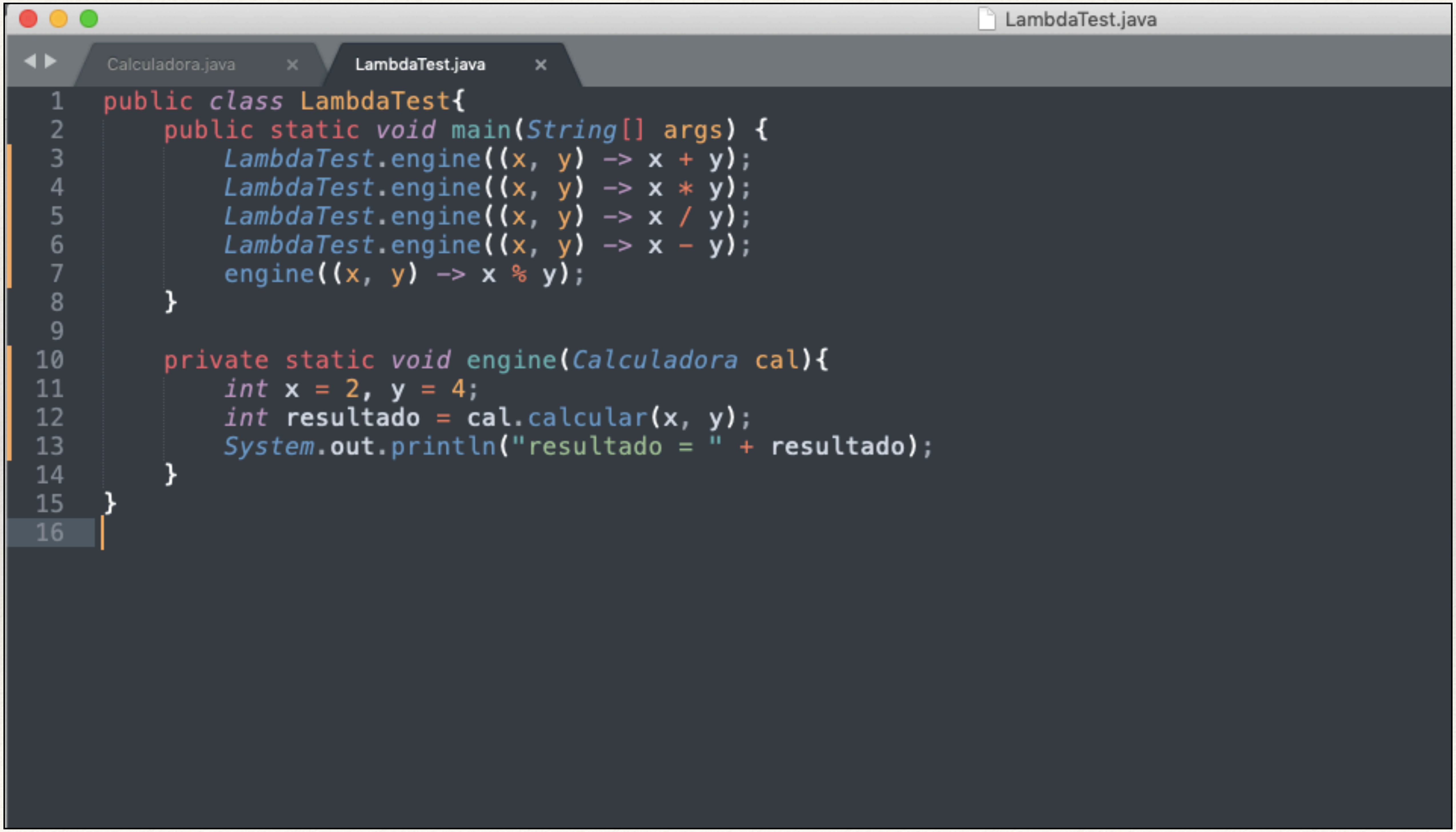
Expresiones lambda con devolución



```java
/*
    Interfaz Funcional
*/

public interface Calculadora{
    public int calcular(int x, int y);
}
```

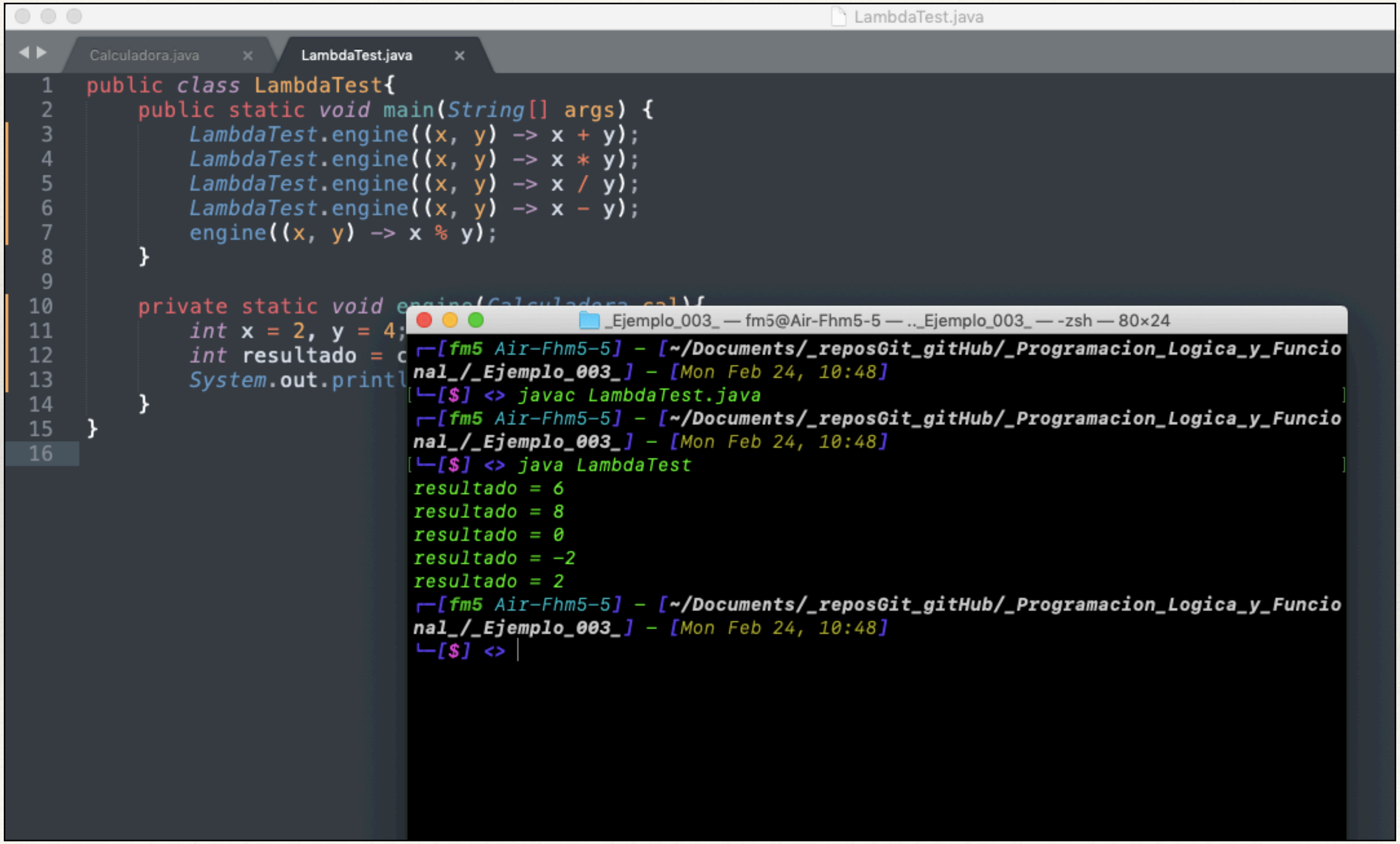
Expresiones lambda con devolución



```java
public class LambdaTest{
    public static void main(String[] args) {
        LambdaTest.engine((x, y) -> x + y);
        LambdaTest.engine((x, y) -> x * y);
        LambdaTest.engine((x, y) -> x / y);
        LambdaTest.engine((x, y) -> x - y);
        engine((x, y) -> x % y);
    }

    private static void engine(Calculadora cal){
        int x = 2, y = 4;
        int resultado = cal.calcular(x, y);
        System.out.println("resultado = " + resultado);
    }
}
```

Expresiones lambda con devolución


LambdaTest.java
Calculadora.java
LambdaTest.java

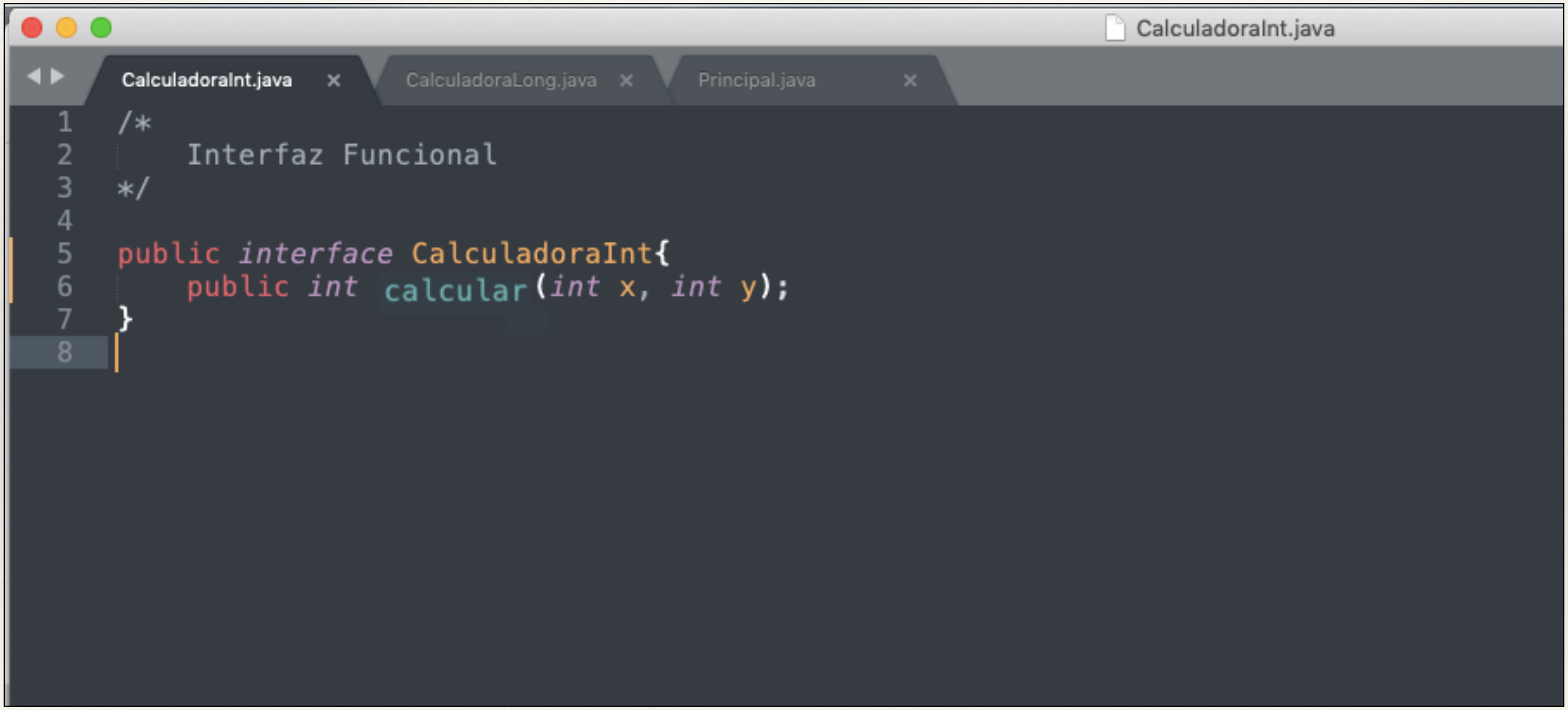
```java
public class LambdaTest{
    public static void main(String[] args) {
        LambdaTest.engine((x, y) -> x + y);
        LambdaTest.engine((x, y) -> x * y);
        LambdaTest.engine((x, y) -> x / y);
        LambdaTest.engine((x, y) -> x - y);
        engine((x, y) -> x % y);
    }

    private static void engine(Calculadora cal){
        int x = 2, y = 4;
        int resultado = c
        System.out.printl
    }
}
```

_Ejemplo_003_ — fm5@Air-Fhm5-5 — .._Ejemplo_003_ — -zsh — 80×24

```
┌─[fm5 Air-Fhm5-5] - [~/Documents/_reposGit_gitHub/_Programacion_Logica_y_Funcio
nal_/_Ejemplo_003_] - [Mon Feb 24, 10:48]
└─[$] <> javac LambdaTest.java
┌─[fm5 Air-Fhm5-5] - [~/Documents/_reposGit_gitHub/_Programacion_Logica_y_Funcio
nal_/_Ejemplo_003_] - [Mon Feb 24, 10:48]
└─[$] <> java LambdaTest
resultado = 6
resultado = 8
resultado = 0
resultado = -2
resultado = 2
┌─[fm5 Air-Fhm5-5] - [~/Documents/_reposGit_gitHub/_Programacion_Logica_y_Funcio
nal_/_Ejemplo_003_] - [Mon Feb 24, 10:48]
└─[$] <>
```

Ambigüedad de tipo en expresiones lambda

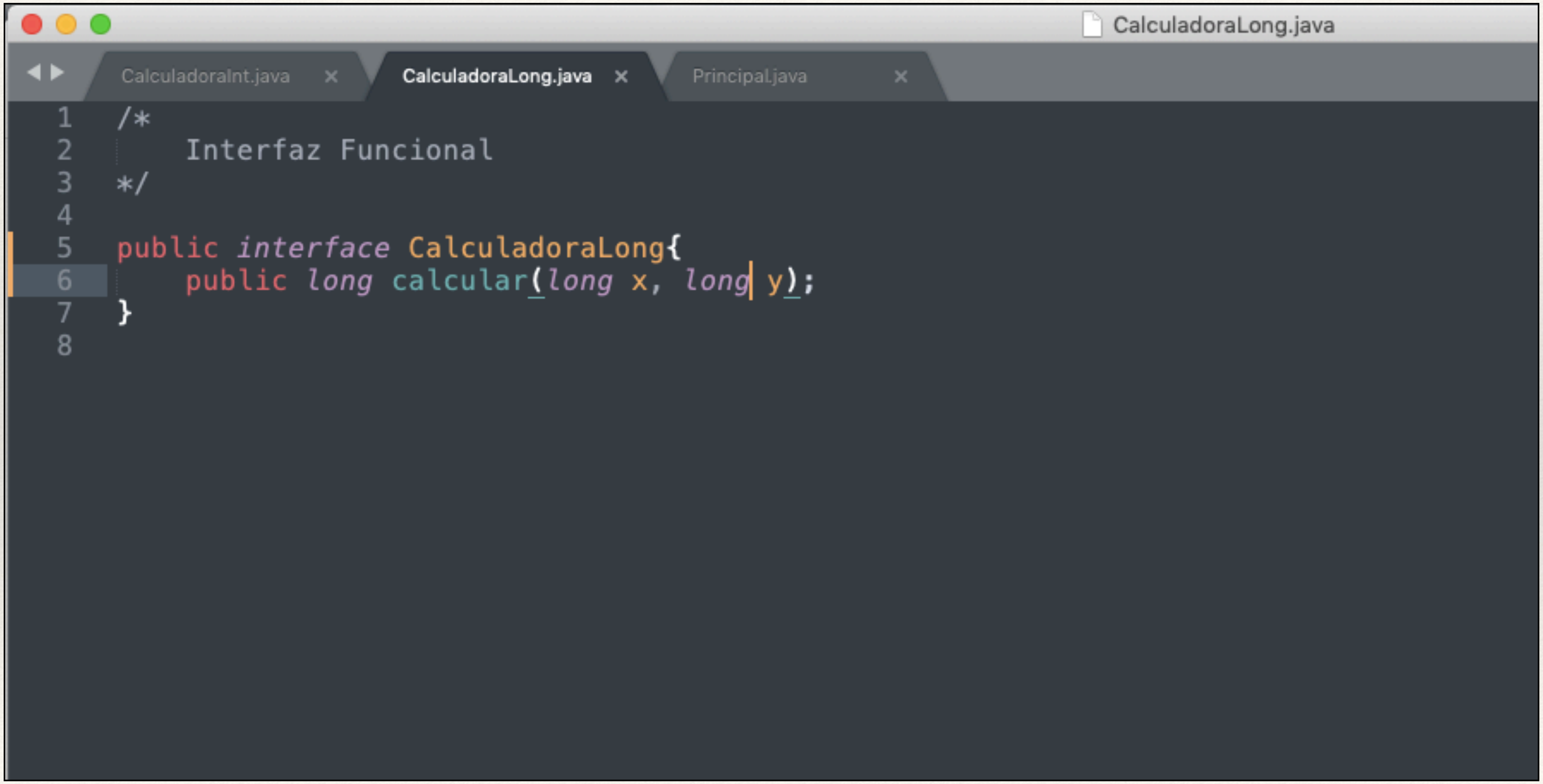


```
/*
    Interfaz Funcional
*/

public interface CalculadoraInt{
    public int calcular(int x, int y);
}

```

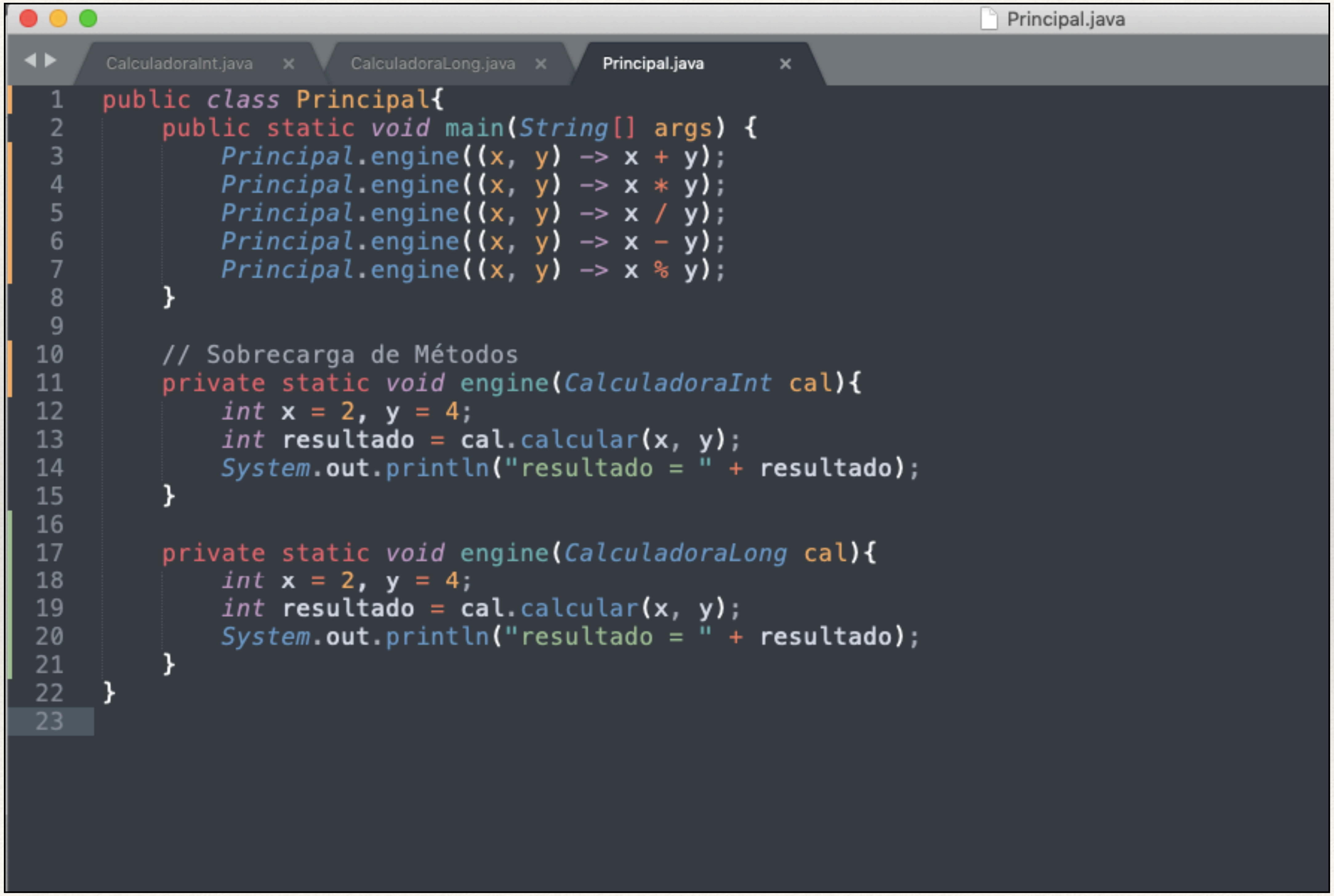
Ambigüedad de tipo en expresiones lambda



```java
/*
    Interfaz Funcional
*/

public interface CalculadoraLong{
    public long calcular(long x, long y);
}
```

Ambigüedad de tipo en expresiones lambda

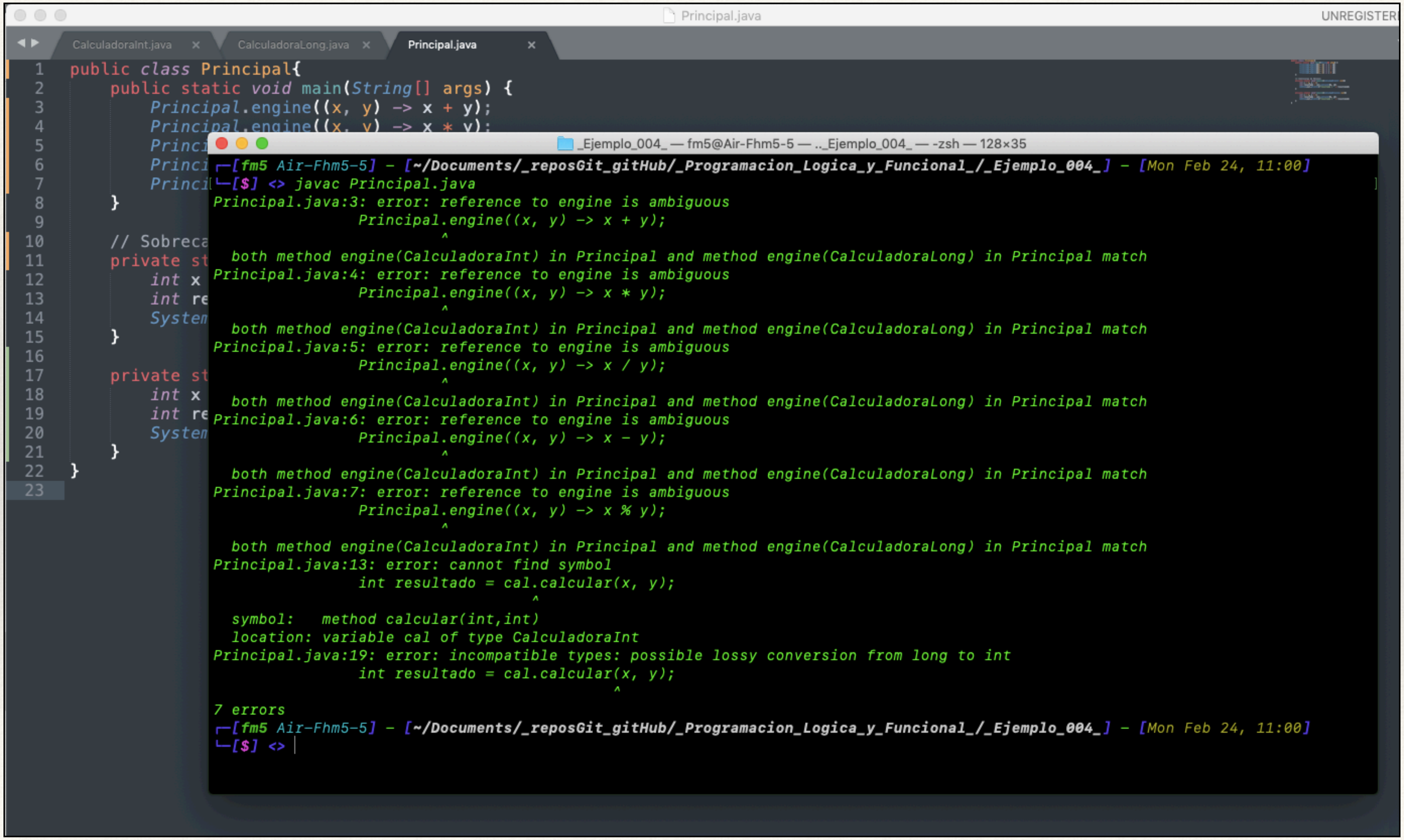


```java
public class Principal{
    public static void main(String[] args) {
        Principal.engine((x, y) -> x + y);
        Principal.engine((x, y) -> x * y);
        Principal.engine((x, y) -> x / y);
        Principal.engine((x, y) -> x - y);
        Principal.engine((x, y) -> x % y);
    }

    // Sobrecarga de Métodos
    private static void engine(CalculadoraInt cal){
        int x = 2, y = 4;
        int resultado = cal.calcular(x, y);
        System.out.println("resultado = " + resultado);
    }

    private static void engine(CalculadoraLong cal){
        int x = 2, y = 4;
        int resultado = cal.calcular(x, y);
        System.out.println("resultado = " + resultado);
    }
}
```

Ambigüedad de tipo en expresiones lambda



```java
public class Principal{
    public static void main(String[] args) {
        Principal.engine((x, y) -> x + y);
        Principal.engine((x, y) -> x * y);
        Princi
        Princi
        Princi
    }

    // Sobreca
    private st
        int x
        int re
        System
    }

    private st
        int x
        int re
        System
    }
}
```

```
┌─[fm5 Air-Fhm5-5] - [~/Documents/_reposGit_gitHub/_Programacion_Logica_y_Funcional_/_Ejemplo_004_] - [Mon Feb 24, 11:00]
└─[$] <> javac Principal.java
Principal.java:3: error: reference to engine is ambiguous
                Principal.engine((x, y) -> x + y);
                         ^
  both method engine(CalculadoraInt) in Principal and method engine(CalculadoraLong) in Principal match
Principal.java:4: error: reference to engine is ambiguous
                Principal.engine((x, y) -> x * y);
                         ^
  both method engine(CalculadoraInt) in Principal and method engine(CalculadoraLong) in Principal match
Principal.java:5: error: reference to engine is ambiguous
                Principal.engine((x, y) -> x / y);
                         ^
  both method engine(CalculadoraInt) in Principal and method engine(CalculadoraLong) in Principal match
Principal.java:6: error: reference to engine is ambiguous
                Principal.engine((x, y) -> x - y);
                         ^
  both method engine(CalculadoraInt) in Principal and method engine(CalculadoraLong) in Principal match
Principal.java:7: error: reference to engine is ambiguous
                Principal.engine((x, y) -> x % y);
                         ^
  both method engine(CalculadoraInt) in Principal and method engine(CalculadoraLong) in Principal match
Principal.java:13: error: cannot find symbol
                int resultado = cal.calcular(x, y);
                                   ^
  symbol:   method calcular(int,int)
  location: variable cal of type CalculadoraInt
Principal.java:19: error: incompatible types: possible lossy conversion from long to int
                int resultado = cal.calcular(x, y);
                                            ^
7 errors
┌─[fm5 Air-Fhm5-5] - [~/Documents/_reposGit_gitHub/_Programacion_Logica_y_Funcional_/_Ejemplo_004_] - [Mon Feb 24, 11:00]
└─[$] <>
```

Ambigüedad de tipo en expresiones lambda

¿Cómo resolvemos el problema (error) anterior?