

Programación funcional II

Ricardo Pérez López

IES Doñana, curso 2019/2020

Índice general

1. Abstracciones funcionales	2
1.1. Definición de funciones anónimas	2
1.2. Parámetros y argumentos	2
1.3. Paso de argumentos por valor	2
1.4. Ámbito de las variables	2
1.5. La sentencia <code>return</code>	2
2. Composición de funciones	2
3. Computabilidad	2
3.1. Funciones recursivas	2
3.2. Un lenguaje Turing-completo	2
4. Tipos de datos compuestos	2
4.1. Las cadenas como datos compuestos	2
4.2. Los <i>arrays</i> como listas inmutables de elementos	2
5. Funciones de orden superior	2
5.1. <code>array_map()</code>	2
5.2. <code>array_filter()</code>	2
5.3. <code>array_reduce()</code>	2
5.4. Funciones locales a funciones	2
5.5. Funciones anónimas	2
6. <i>Scripts</i>	2

1. Abstracciones funcionales

1.1. Definición de funciones anónimas

1.2. Parámetros y argumentos

1.3. Paso de argumentos por valor

1.4. Ámbito de las variables

1.5. La sentencia `return`

2. Composición de funciones

3. Computabilidad

3.1. Funciones recursivas

3.2. Un lenguaje Turing-completo

4. Tipos de datos compuestos

4.1. Las cadenas como datos compuestos

4.2. Los *arrays* como listas inmutables de elementos

5. Funciones de orden superior

5.1. `array_map()`

5.2. `array_filter()`

5.3. `array_reduce()`

5.4. Funciones locales a funciones

5.5. Funciones anónimas

6. *Scripts*