

Semana 2 – Grupo Teorico

Los POOsexuales

Escuela de Ingeniería en Sistemas
poo@ues.edu.sv

Comunismo

Universidad de El Salvador
comunismo@ues.edu.sv

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent convallis orci arcu, eu mollis dolor. Aliquam eleifend suscipit lacinia. Maecenas quam mi, porta ut lacinia sed, convallis ac dui. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse potenti.

2012 ACM Subject Classification Replace ccsdesc macro with valid one, e.g., ‘Information systems → Question answering’

Keywords and phrases keyword one, keyword 2, three

Digital Object Identifier 10.4230/LIPICs.CVIT.2023.1

Tabla de Contenidos

1	Clase Lunes	2
1.1	Especificación semiformal	2
1.2	Implementación de los TAD	2
1.3	Ejercicio 1	3
2	Clase Miercoles	3
2.1	Ejercicio 2	3



© Camilo Medrano;
licensed under Creative Commons License CC-BY
Clases Teoricas de ESD115 - 2023.

Editor: Camilo Medrano; Article No. 1; pp. 1:1–1:3



Leibniz International Proceedings in Informatics
LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

23 **1 Clase Lunes**

24 struct: implementación física de un TAD.

25 No hay reglas para el nombre del TAD.

26 ► **Ejemplo 1.** Captura 15:44

27 → Hay un material en Word.

28 **1.1 Especificación semiformal**

Definición de valor	Definición de operación
Clausula de definición	Encabezado
Clausula de condición	Condiciones previas
	Coniciones posteriores

29 ► **Comentario 2.** ■ Clausula de condición: Debe cumplir para pertenecer.

30 ■ Condiciones posteriores: Que va a realizar con los argumentos que tiene.

31 ► **Ejemplo 3.** Captura 15:23■ **Listing 1** Forma 2

```

32
33  /* Comentario
34  multilinea */
35
36  // Comentario de una sola linea
37

```

38 typedef → definición de un tipo en Java y en C.

39 ► **Nota 4.** Se esta siguiendo la sintaxis de C.40 ► **Ejemplo 5.** Otro ejemplo en el archivo.■ **Listing 2** Forma 2

```

41
42  Las secuencias: pag 8
43

```

44 **1.2 Implementación de los TAD**

45 Cap 15:47

46 Cap 15:50

47 ► **Comentario 6.** C no es orientado a objetos es estructurado.48 ► **Nota 7.** En las prácticas se trabaja con C++

49 1.3 Ejercicio 1

50 Se quiere diseñar un tipo abstracto de dato que represente a una fecha en el formato (dd, mm,
51 aa). Debe tener las siguientes operaciones: crear una fecha, inicializar una fecha, modificar
52 el día, modificar el mes, modificar el año, comparar 2 fechas, desplegar la fecha.

■ Listing 3 Definición de tipo

```
53 abstract typedef <integer, 3> FECHA;  
54 condition FECHA[0]>0 && FECHA[0]<32  
55           FECHA[1]>0 && FECHA[1]<13  
56           FECHA[2]>=0  
57 // Ella lo hizo sin condición  
58  
59
```

60 ► Nota 8. ¿La clausula de condición para que sirve?

■ Listing 4 Definición de crear

```
61 abstract FECHA create(); // Crearla con valores predefinidos  
62 postcondition create[0]=29;  
63           create[1]=11;  
64           create[2]=2004;  
65  
66
```

■ Listing 5 Forma 2

```
67 abstract FECHA inicialice(FECHA f, int day, int month, int year)  
68 precondition // validacion de fecha  
69 postcondition f[0]=day  
70           f[1]=month  
71           f[2]=year  
72  
73
```

74 2 Clase Miercoles

75 ► Comentario. Otros métodos del ejercicio anterior

76 2.1 Ejercicio 2

77 Se quiere diseñar un tipo abstracto de dato que represente un instrumento musical, del cual se
78 conoce un código numérico, un nombre y un tipo (1- de cuerda, 2- de viento, 3- de percusión,
79 4- de teclas). Debe tener las siguientes operaciones: crear un instrumento, inicializar un
80 instrumento, modificar el código, modificar el nombre, comparar dos instrumentos.