

[Página Principal](#) / [Mis cursos](#) / [2022-2K4-4-229](#) / [Semana 14](#) / [SEGUNDO PARCIAL PRÁCTICO - 1 CUATRIMESTRE 2022](#)

<b>Comenzado el</b>	sábado, 18 de junio de 2022, 08:30
---------------------	------------------------------------

<b>Estado</b>	Finalizado
---------------	------------

<b>Finalizado en</b>	sábado, 18 de junio de 2022, 09:08
----------------------	------------------------------------

<b>Tiempo empleado</b>	38 minutos 32 segundos
----------------------------	------------------------

Pregunta 1

Finalizado

Puntúa como 100,00

## Segundo Parcial

### Paradigma de Programación

#### Objetivo

Evaluar al estudiante en la parte práctica de las unidades nro. 4 y nro. 5 (Paradigma Funcional y Paradigma Lógico, respectivamente) a partir de la resolución que guarde en los archivos más abajo especificados, correspondientes a las consignas solicitadas para los ejercicios de cada paradigma.

#### Condiciones de trabajo:

- El día pactado para el segundo parcial en el horario informado oportunamente, en el aula virtual de Moodle estará habilitada una actividad tipo cuestionario identificada como "2 Parcial Práctico". El estudiante deberá ingresar al cuestionario que consiste de una sola consigna, con el enunciado que deberá resolver.
- Este parcial práctico consta de dos partes: programación lógica y programación funcional. Para cada paradigma se deberá desarrollar un programa, utilizando el correspondiente entorno.
- Para resolver las consignas correspondientes al paradigma lógico, se deberá generar un archivo con el nombre **Legajo\_ApellidoNombre.pl** para definir los hechos y reglas que se soliciten en las consignas que se detallan más abajo.
- Para resolver las consignas correspondientes al paradigma funcional, se deberá generar un archivo con el nombre **Legajo\_ApellidoNombre.hs** para formular las funciones que se soliciten en las consignas que se detallan más abajo.
- La duración prevista para el parcial es de 1:30hs, y comenzará en el horario asignado,
- Al finalizar el desarrollo del programa, el estudiante deberá subir los archivos .hs y .pl al cuestionario correspondiente al parcial, dentro de la hora de finalización especificada.
- Es responsabilidad de cada estudiante ir guardando periódicamente su archivo de trabajo, como así también del contenido de los mismos.
- ACLARACION: El estudiante durante el transcurso del examen, no podrá tener dos instancias de UV abiertas y no puede salir del cuestionario sin finalizarlo correctamente, ya que la evaluación quedará invalidada.

### Paradigma Lógico

#### Enunciado

Durante la semana patria del 25 de mayo un cine ofrece proyectar películas infantiles a precios promocionales, para todo el público. El cine dispone de 4 salas, y su objetivo con estas promociones es llenar la capacidad de cada una de las salas durante cada función.

Para cada función se establece una tarifa por la entrada, y toda entrada para dicha función se vende a esa tarifa. Cada entrada da derecho a ocupar una butaca numerada del cine.

A continuación, se muestra la **tabla 1** que contiene los datos registrados de las diferentes funciones brindadas durante esa semana. De cada función se registra su código, el código de la película proyectada, la fecha de la función, el listado de números de butacas que no fueron vendidas para esa función, el importe de la tarifa única por entrada, y el código de la sala en la que se realizó la función.

**Tabla 1: Funciones**

Código de función	Código de película	fecha			Listado de butacas no vendidas en la sala	Tarifa por entrada [\$]	Código de sala
		Día	Mes	Año			
fun1	pe13	21	5	2022	[12, 20, 30]	200	sal1
fun2	pe11	21	5	2022	[1, 10]	220	sal2
fun3	pe11	22	5	2022	[2, 12, 14, 20, 23]	180	sal3
fun4	pe12	22	5	2022	[ ]	220	sal4
fun5	pe13	23	5	2022	[20, 21, 25]	150	sal1
fun6	pe12	23	5	2022	[30, 31]	180	sal2

fun7	pel2	24	5	2022	[4, 18, 19, 30, 33]	220	sal1
fun8	pel1	25	5	2022	[ ]	300	sal2
fun9	pel2	25	5	2022	[ ]	300	sal3
fun10	pel1	25	5	2022	[ ]	300	sal4

La **tabla 2** corresponde a las diferentes salas del cine. De cada sala se registra su código, la descripción de la misma, si dispone de proyección 3D o no, y la capacidad, es decir, de la cantidad total de entradas que se pueden vender para llenar la sala.

**Tabla 2: Salas**

Código de sala	Descripción	Proyector 3D	Cantidad total de butacas (capacidad)
sal1	'Sala 1 - 3D - 50'	si	50
sal2	'Sala 2 - común - 40'	no	40
sal3	'Sala 3 - común - 30'	no	30
sal4	'Sala 4 - 3D - 40'	si	40

La **tabla 3** corresponde a las películas que se promocionan durante la semana del 25 de mayo. De cada película se registra su código, título y el año en el que se filmó.

**Tabla 3: Películas.**

Código de película	Título	Año de filmación
pel1	'La Dama y el Vagabundo'	1997
pel2	'100 Dálmatas'	2001
pel3	'Pluto y sus Amigos'	2019
pel4	'Aristoperros'	2020

A partir de los hechos que se detallan a continuación, que representan todos los datos de las tablas 1,2 y 3.

#### Hechos:

función(fun1,pel3,fecha(21,5,2022),[12, 20, 30],200,sal1).

función(fun2,pel1,fecha(21,5,2022),[1, 10],220,sal2).

función(fun3,pel1,fecha(22,5,2022),[2, 12, 14, 20, 23],180,sal3).

función(fun4,pel2,fecha(22,5,2022),[ ],220,sal4).

función(fun5,pel3,fecha(23,5,2022),[20, 21, 25],150,sal1).

función(fun6,pel2,fecha(23,5,2022),[30, 31],180,sal2).

función(fun7,pel2,fecha(24,5,2022),[4, 18, 19, 30, 33],220,sal1).

función(fun8,pel1,fecha(25,5,2022),[ ],300,sal2).

función(fun9,pel2,fecha(25,5,2022),[ ],300,sal3).

función(fun10,pel1,fecha(25,5,2022),[ ],300,sal4).

sala(sal1,'Sala 1 - 3D - 50',si,50).

sala(sal2,'Sala 2 – común - 40',no,40).

sala(sal3,'Sala 3 – común - 30',no,30).

sala(sal4,'Sala 4 - 3D - 40',si,40).

película(peli1,'La Dama y el Vagabundo',1997).

película(peli2,'100 Dálmatas',2001).

película(peli3,'Pluto y sus Amigos',2019).

película(peli4,'Aristoperros',2020).

### Requerimientos:

Usted deberá definir las reglas que permitan resolver lo siguiente:

**1.** Conocer si entre las funciones del día 25/05/2022 se proyectó o no, alguna de las dos películas cuyos correspondientes códigos de películas se reciben como argumentos. Predicado sugerido para esta regla: regla1/2. **(15 puntos)**

Ejemplos:

1?- regla1(peli1,peli2).

true.

2?- regla1(peli2,peli3).

true.

3?- regla1(peli3,peli4).

false.

**2.** Conocer a partir del código de una función, el título de la película proyectada y la descripción completa de la sala correspondiente. Predicado sugerido para esta regla: regla2/3. **(15 puntos)**

Ejemplos:

1?- regla2(fun1,TítuloPeli,DescripSala).

TítuloPeli = 'Pluto y sus Amigos',

DescripSala = 'Sala 1 - 3D - 50'.

2?- regla2(fun2,TítuloPeli,DescripSala).

TítuloPeli = 'La Dama y el Vagabundo',

DescripSala = 'Sala 2 – común - 40'.

3?- regla2(fun4,TítuloPeli,DescripSala).

TítuloPeli = '100 Dálmatas',

DescripSala = 'Sala 4 - 3D - 40'.

**3.** Generar una lista con los códigos de funciones para las cuales no se vendieron una cantidad de entradas mayor o igual a cierta cantidad numérica que se especifica como primer argumento de la regla. Predicado sugerido para esta regla: regla3/2. **(20 puntos)**

Ejemplos:

1?- regla3(5,Lista).

Lista = [fun3, fun7].

2?- regla3(3,Lista).

Lista = [fun1, fun3, fun5, fun7].

3?- regla3(2,Lista).

Lista = [fun1, fun2, fun3, fun5, fun6, fun7].

**Enunciado**

El mismo cine que promociona películas infantiles nos ha solicitado además el desarrollo de un programa en Haskell.

**Requerimientos:**

Usted deberá definir las funciones que permitan resolver lo siguiente:

**1.** Realizar una función que reciba el código de una sala y retorne la capacidad, es decir la cantidad máxima de asientos que se pueden llegar a ocupar para ver alguna película allí, tal como se especifica en la tabla 1. En el caso de que el código de la sala no se encuentre especificado en la tabla 1, la función deberá retornar el valor 0. **(15 puntos)**

**Tabla 1: Capacidad máxima de personas por sala**

Código de sala	Capacidad
"sal1"	50
"sal2"	40
"sal3"	30
"sal4"	40

Ejemplos:

```
Main> funcion1 "sal1"
```

```
50
```

```
Main> funcion1 "sal2"
```

```
40
```

```
Main> funcion1 "sal3"
```

```
30
```

```
Main> funcion1 "sala"
```

```
0
```

**2.** Realizar una función que reciba 2 parámetros: una lista de códigos de salas utilizadas para diferentes funciones, y como segundo parámetro, una única tarifa fija. La función deberá generar una nueva lista con los importes máximos para cada una de las funciones que utilizaron las diferentes salas especificadas en la lista recibida como primer parámetro. Cada uno de los importes de la lista a generar se deberá calcular de la siguiente manera: capacidad de la correspondiente sala multiplicado por la tarifa fija única que se recibe como segundo argumento. **(15 puntos)**

Ejemplos:

```
Main> funcion2 ["sal1","sal2","sal1","sal3"] 100
```

```
[5000,4000,5000,3000]
```

```
Main> funcion2 ["sal1","sal2","sal1","sal3"] 50
```

```
[2500,2000,2500,1500]
```

```
Main> funcion2 ["sal1","sal2","sal1","sal3"] 200
```

```
[10000,8000,10000,6000]
```

**3.** Realizar una función que reciba 2 parámetros: una lista de códigos de salas utilizadas para diferentes funciones, y como segundo parámetro, un valor numérico. La función deberá determinar la cantidad de capacidades de salas asociadas a cada uno de los códigos de salas especificados en la lista que se recibe como primer parámetro, mayores o iguales al valor numérico recibido como segundo parámetro. Para resolver este punto debe definir una función recursiva. **(20 puntos)**

Ejemplos:

```
Main> funcion3 ["sal1","sal2","sal1","sal3"] 20
```

```
4
```

```
Main> funcion3 ["sal1","sal2","sal1","sal3"] 40
```

```
3
```

Main> funcion3 ["sal1","sal2","sal1","sal3"] 50

2

### Tabla de valoración de los ítems evaluados

Ítems a Evaluar	Pje	Observaciones	Obtenido
Implementación de la regla 1.	15		
Implementación de la regla 2.	15		
Implementación de la regla 3.	20		
Implementación de la función 1.	15		
Implementación de la función 2.	15		
Implementación de la función 3.	20		
Total	100		

 [\\_90328\\_WedemeyerFederico.pl](#)

 [\\_90328\\_WedemeyerFederico.hs](#)

Comentario:

- 1) No hacen falta dos cortes. Se puede simplificar la regla.
- 3) Son codigos, deberia mostrar la lista sin repeticiones despues de usar el findall.

-

Faltan los tipos de las funciones.

- 2) La tarifa debe ser un numero positivo.

◀ [Cuestionario 5. Unidad nro. 5: paradigma de programación lógica](#)

Ir a...

[EjerciciosAdicionales-Enunciados-Repaso](#) ►