

Dadas las estructuras de datos adjuntas sobre las sesiones de los multicines de Málaga en la semana actual y la información sobre las distintas películas, diseñar dos subprogramas que:

- ```
/*
 * Salas de cines de Málaga.
 */
CONST
 N MaxPelículas = 200 // en Málaga
 N MaxProyCine = 100 // para un multicine en un día
 N MaxCadena = 256
```

```

ENUM {Lun, Mar, Mie, Jue, Vie, Sab, Dom} TDiaSem
ENUM {Alameda, Albeniz, America, Rosaleda} TCine
ENUM {TP, M13, M18} TPublico
C TCadena[0..MaxCadena-1]
REGISTRO THora
 N hora, min
FINREGISTRO

```

```
// número de salas de cada multicine
N NSalas[Alameda..Rosaleda] = { 3, 4, 7, 10 }
```

```
THora comienzo
N cod_peli // 0 -> no proyección
N sala // en el rango adecuado para cada cine
```

```
UnaProyeccion TProyecciones[1..MaxProyCine]
```

```
TProyecciones TCartelera[Alameda..Rosaleda][Lun..Dom]
```

|     | Alameda |    | Alvariz |    | America |    | Bogalada |    |    |    |     |   |  |  |  |  |  |  |  |  |
|-----|---------|----|---------|----|---------|----|----------|----|----|----|-----|---|--|--|--|--|--|--|--|--|
|     | 12      | 2  | 18      | 20 | 0       | 0  | 12       | 45 | 23 | 3  |     |   |  |  |  |  |  |  |  |  |
| Tun | 13      | 20 | 123     | 2  | 16      | 30 | 0        | 0  | 20 | 45 | 23  | 3 |  |  |  |  |  |  |  |  |
| Mar | 12      | 45 | 126     | 3  | 16      | 30 | 231      | 4  | 18 | 45 | 163 | 1 |  |  |  |  |  |  |  |  |
| Mie | 13      | 30 | 54      | 0  | 18      | 30 | 24       | 3  | 21 | 45 | 172 | 2 |  |  |  |  |  |  |  |  |
| Tue | 14      | 40 | 123     | 1  | 17      | 30 | 0        | 0  | 20 | 45 | 231 | 3 |  |  |  |  |  |  |  |  |
| Vie | 16      | 05 | 67      | 5  | 14      | 30 | 0        | 0  | 20 | 10 | 23  | 6 |  |  |  |  |  |  |  |  |
| Sab | 12      | 00 | 231     | 6  | 15      | 30 | 45       | 3  | 20 | 45 | 131 | 3 |  |  |  |  |  |  |  |  |
| Dom | 15      | 20 | 123     | 2  | 18      | 30 | 0        | 0  | 22 | 00 | 173 | 1 |  |  |  |  |  |  |  |  |

```

REGISTRO UnaPelicula
 N codigo // único para cada película 0 --> hueco
 TCadena titulo, prota1, prota2
 TPublico publico
 N duracion // en minutos
FINREGISTRO
UnaPelicula TPeliculas[1..MaxPeliculas]

```

| codigo | titulo         | protal        | prota2        | publico | duracion |
|--------|----------------|---------------|---------------|---------|----------|
| 231    | "Jarri Pota"   | "Jacinto Pi"  | "Pepa Guti"   | TP      | 92       |
| 67     | "Sr. anillo"   | "Lolo Jones"  | "Mary Perez"  | TP      | 178      |
| 13     | "SIn noticias" | "Vicky Mayo"  | "Lope Cruise" | M18     | 94       |
| 173    | "Gran familia" | "Beto Closas" | "Loli Camus"  | TP      | 118      |
| ...    | ...            | ...           | ...           | ...     | ...      |

```

/*
 * Escribe la cadena de caracteres correspondiente
 */
ALGORITMO EscribirDia(E TDiaSem dsem)
INICIO
 CASO dsem SEA
 Lun: Escribir("lunes")
 Mar: Escribir("martes")
 Mie: Escribir("miércoles")
 Jue: Escribir("jueves")
 Vie: Escribir("viernes")
 Sab: Escribir("sábado")
 Dom: Escribir("domingo")
 FINCASO
FIN

/*
 * Escribe la cadena de caracteres correspondiente
 */
ALGORITMO EscribirCine(E TCine cine)
INICIO
 CASO cine SEA
 Alameda: Escribir("Alameda")
 Albeniz: Escribir("Albeniz")
 America: Escribir("America")
 Rosaleda: Escribir("Rosaleda")
 FINCASO
FIN

/*
 * Devuelve el índice de la película "codigo" en la
 * estructura de las películas si está, ó 0 si no.
 */
ALGORITMO N BuscarPelicula(E TPeliculas peliculas; E N codigo)
VAR N j; B encontrado
INICIO
 encontrado = FALSO; j=1
 MIENTRAS (j<= MaxPeliculas) Y NO encontrado HACER
 SI peliculas[j].codigo==codigo ENTONCES
 encontrado = VERDADERO
 SINO
 j = j+1
 FINSI
 SI NO encontrado ENTONCES
 j = 0
 FINSI
 DEVOLVER j
FIN

```

```

/*
 * Muestra título de una película y el horario de
 * su proyección.
 */
ALGORITMO MostrarProyeccion(E TCadena titulo; E TDiaSem dia; E THora inicio)
VAR
 N i
INICIO
 Escribir("Título: ", titulo)
 EscribirDia(dia)
 Escribir(" ", inicio.hora, ":", inicio.min)
FIN

/*
 * Muestra las proyecciones de todas las películas para todos los públicos.
 */
ALGORITMO Mostrar_TP(E TCartelera cartel; E TPeliculas pelis)
VAR
 TCine cine
 TDiaSem dia
 N i, j, codigo, sala
INICIO
 PARA cine=Alameda HASTA Rosaleda HACER
 Escribir("Cine: "); EscribirCine(cine)
 PARA dia=Lun HASTA Dom HACER
 PARA i=1 HASTA MaxProyCine HACER
 codigo = cartel[cine][dia][i].cod_peli
 SI codigo!=0 ENTONCES
 sala = cartel[cine][DIA][i].sala
 j = BuscarPelicula(pelis, codigo)
 SI sala<=NSalas[cine] Y j != 0 Y pelis[j].publico==TP ENTONCES
 Escribir("Sala ", sala)
 MostrarProyeccion(pelis[j].titulo, dia,
 cartel[cine][dia][i].comienzo)
 FINSI
 FINSI
 FINPARA
 FINPARA
FIN

/*
 * Devuelve la hora de finalización de una película.
 * Devuelve por convenio la hora 25:00 si la película no está en cartelera.
 */
ALGORITMO THora HoraFin(E TPeliculas pelis; E UnaProyeccion proy)
VAR
 N i, minutos
 THora fin
INICIO
 i = BuscarPelicula(pelis, proy.cod_peli)
 SI i == 0 ENTONCES
 fin.hora = 25; fin.min = 0
 SINO
 minutos = pelis[i].duracion
 fin.min = (proy.comienzo.min + minutos) MOD 60
 fin.hora = proy.comienzo.hora + (proy.comienzo.min + minutos) DIV 60
 FINSI
 DEVOLVER fin
FIN

```

```

/*
 * Devuelve VERDADERO si la primera hora es más temprana que la segunda.
 */
ALGORITMO B Antes(E THora h1, h2)
INICIO
 DEVOLVER h1.hora<h2.hora O (h1.hora==h2.hora Y h1.min<h2.min)
FIN

/*
 * Escribe título, hora comienzo y finalización de
 * las sesiones de un día que acaban antes de una determinada hora.
 */
ALGORITMO EscribirNoGolfas(E TPeliculas pelis; E TProyecciones sesiones;
 E TDiaSem dia; E THora fin)
VAR
 N codigo, i, j
 TCine cine
 THora finsesion
INICIO
 PARA cine=Alameda HASTA Rosaleda HACER
 PARA i=1 HASTA MaxProyCine HACER
 codigo = sesiones[cine][dia][i].cod_peli
 SI codigo != 0 ENTONCES
 j = BuscarPelícula(pelis, codigo)
 SI j != 0 ENTONCES // ¿NECESARIO?
 finsesion = HoraFin(pelis,sesiones[cine][dia][i])
 SI Antes(finsesion, fin) ENTONCES
 MostrarProyeccion(pelis[i].titulo, dia,
 sesiones[cide][dia][i].comienzo)
 FINSI
 FINSI
 FINSI
 FINPARA
 FIN
FIN

```

Modularización:

