

# **Práctica 2**

Fundamentos de Bases de datos: Shaofan Xu y Alejandro Zheng

## 1. OBJETIVO

El objetivo de esta práctica es desarrollar un programa en C que acceda a la base de datos flights mediante la biblioteca ODBC.

El programa ofrece una interfaz de texto basada en ncurses que permita al usuario:

- a. Buscar vuelos entre dos aeropuertos, incluyendo vuelos con una conexión.
- b. Emitir tarjetas de embarque para las reservas existentes, asignando asientos disponibles automáticamente.

Con esta práctica hemos aprendido cómo conectar un programa en C con una base de datos, ejecutar consultas SQL de forma segura usando prepared statements, y gestionar correctamente la memoria y los resultados obtenidos.

## 2. IMPLEMENTACIÓN

### a. REALIZACIÓN DE BÚSQUEDA

Para la implementación de esta consulta, lo primero que hemos hecho es crear dos tablas diferentes una para los vuelos directos y otra para los vuelos con transbordo con sintaxis WITH. Para la tabla de vuelos directos hemos puesto que la conexión es igual a 0 puesto que no está conectado con otros vuelos. En la consulta hemos asignado que el horario de partida tiene que ser el mismo que hemos puesto en DATE, el aeropuerto de salida y llegada es igual a FROM y TO. Luego para calcular los asientos libres que están disponibles primero hemos contado todos los asientos que están reservados con un COUNT, después con un NOT IN excluimos los asientos reservados y obtenemos los asientos libres.

Para la tabla de vuelos con transbordo hacemos de forma parecida para los asientos libres, hemos puesto la conexión es igual a 1 ya que existe conexión entre los vuelos. Además, en esta tabla tenemos que fijar que la llegada de primer vuelo tiene que ser antes que la salida de segundo vuelo ( $f1.scheduled\_arrival < f2.scheduled\_departure$ ) y parte de que la salida de primer vuelo tiene que ser mismo día que DATE, también hemos puesto la duración total de viaje no puede superar 24 horas ( $f2.scheduled\_arrival - f1.scheduled\_departure \leq '24 hours'$ ).

Finalmente hacemos una unión de las dos tablas con la condición de que el número de asientos libres tiene que ser mayor que 0 y los resultados están ordenados por la duración total de viaje con un orden ascendente.

## b. EMISIÓN DE TARJETAS DE EMBARQUE

Antes de asignar los asientos, creamos subtablas para facilitar nuestra implementación de código SQL. Primero hemos creado una subtabla **booking\_ticket** donde se guardan book\_ref, ticket\_no, flight\_id y etcétera de aquellas reservas cuyos vuelos no tienen todavía tarjeta de embarque. Y **available\_seats** suyo atributo seat\_no, aircraft\_code, flight\_id y seat\_rank (enumera cada una de las tuplas) donde contiene los asientos vacíos de cada vuelo. Además, **boarding\_passes\_last\_boarding\_no** nos ayuda a asignar número de tarjeta de embarque, cuyos atributos flight\_id y boarding\_no (en nuestro caso el máximo de los números de tarjetas de embarques). Finalmente, **ticket\_to\_allocate** asigna por cada uno de los tipos y flight\_id (que no tienen todavía una tarjeta de embarque) un asiento y un numero de boarding\_passes. Así, en la subtabla **ins** insertamos en boarding\_passes los ticket\_no y flight\_id con sus asientos y número tarjeta de embarque correspondiente. Y devolvemos nombre del pasajero, flight\_id, scheduled\_departure y seat\_no mediante select para el Menú.

En el caso de subtabla **ins** es necesario ya que nos evita a tener problemas de asignar la misma plaza 2 veces usando la sentencia **ON CONFLICT DO NOTHING**.

## c. Paginación del menú

Para ellos hemos usado aritmética de puntero, así definiendo un macro **MAX\_N\_CHOICES** que pasa a los result\_search y result\_bpass en el campo de **max\_rows**, ya que en principio el programa solo pasa el número de columnas que cabe en **out\_window** en una sola página, por lo tanto, también es necesario modificar window:create\_out, para que se crea **MAX\_N\_CHOICES** **out\_win\_choices**.

## 3. EJEMPLOS DE EJECUCIÓN

### a. REALIZACIÓN DE BÚSQUEDA

Las siguientes imágenes son algunos ejemplos de las consultas que hemos realizado.

#### Ejemplo 1:

Para este ejemplo hemos introducido From= DME, To = VKO y Date=2017-09-09.

Out							
2274	2017-09-09 17:40:00	2017-09-09 19:50:00	1	97	02:10:00		
76	2017-09-09 08:35:00	2017-09-09 11:15:00	1	170	02:40:00		
1012	2017-09-09 08:30:00	2017-09-09 13:20:00	1	12	04:50:00		
1012	2017-09-09 08:30:00	2017-09-09 13:35:00	1	12	05:05:00		
4	2017-09-09 11:25:00	2017-09-09 17:10:00	1	170	05:45:00		
234	2017-09-09 10:25:00	2017-09-09 17:10:00	1	170	06:45:00		
76	2017-09-09 08:35:00	2017-09-09 17:10:00	1	170	08:35:00		
196	2017-09-09 18:05:00	2017-09-10 10:10:00	1	170	16:05:00		
196	2017-09-09 18:05:00	2017-09-10 11:15:00	1	170	17:10:00		
2998	2017-09-09 14:20:00	2017-09-10 08:45:00	1	97	18:25:00		
3016	2017-09-09 13:50:00	2017-09-10 08:45:00	1	97	18:55:00		
2998	2017-09-09 14:20:00	2017-09-10 09:55:00	1	97	19:35:00		
3016	2017-09-09 13:50:00	2017-09-10 09:55:00	1	97	20:05:00		
2184	2017-09-09 15:05:00	2017-09-10 11:10:00	1	50	20:05:00		
2998	2017-09-09 14:20:00	2017-09-10 10:30:00	1	97	20:10:00		
3016	2017-09-09 13:50:00	2017-09-10 10:30:00	1	97	20:40:00		
3044	2017-09-09 11:10:00	2017-09-10 08:45:00	1	97	21:35:00		
4	2017-09-09 11:25:00	2017-09-10 10:10:00	1	170	22:45:00		
3044	2017-09-09 11:10:00	2017-09-10 09:55:00	1	97	22:45:00		
196	2017-09-09 18:05:00	2017-09-10 17:10:00	1	170	23:05:00		
3044	2017-09-09 11:10:00	2017-09-10 10:30:00	1	97	23:20:00		
234	2017-09-09 10:25:00	2017-09-10 10:10:00	1	170	23:45:00		
4	2017-09-09 11:25:00	2017-09-10 11:15:00	1	170	23:50:00		

Msg
From= DME , To= VKO , Date= 2017-09-09

En la ventana de Out podemos ver los resultados que obtenemos tras hacer la consulta. Donde la primera columna es flight\_id, luego esta scheduled\_departure, scheduled\_arrival, número de connections, asientos libres y total duración de viaje. Además, si entramos en uno de los resultados concreto, en la ventana de Msg podemos observar los detalles de ese resultado en concreto:

Out							
2274	2017-09-09 17:40:00	2017-09-09 19:50:00	1	97	02:10:00		
76	2017-09-09 08:35:00	2017-09-09 11:15:00	1	170	02:40:00		
1012	2017-09-09 08:30:00	2017-09-09 13:20:00	1	12	04:50:00		
1012	2017-09-09 08:30:00	2017-09-09 13:35:00	1	12	05:05:00		
4	2017-09-09 11:25:00	2017-09-09 17:10:00	1	170	05:45:00		
234	2017-09-09 10:25:00	2017-09-09 17:10:00	1	170	06:45:00		
76	2017-09-09 08:35:00	2017-09-09 17:10:00	1	170	08:35:00		
196	2017-09-09 18:05:00	2017-09-10 10:10:00	1	170	16:05:00		
196	2017-09-09 18:05:00	2017-09-10 11:15:00	1	170	17:10:00		
2998	2017-09-09 14:20:00	2017-09-10 08:45:00	1	97	18:25:00		
3016	2017-09-09 13:50:00	2017-09-10 08:45:00	1	97	18:55:00		
2998	2017-09-09 14:20:00	2017-09-10 09:55:00	1	97	19:35:00		
3016	2017-09-09 13:50:00	2017-09-10 09:55:00	1	97	20:05:00		
2184	2017-09-09 15:05:00	2017-09-10 11:10:00	1	50	20:05:00		
2998	2017-09-09 14:20:00	2017-09-10 10:30:00	1	97	20:10:00		
3016	2017-09-09 13:50:00	2017-09-10 10:30:00	1	97	20:40:00		
3044	2017-09-09 11:10:00	2017-09-10 08:45:00	1	97	21:35:00		
4	2017-09-09 11:25:00	2017-09-10 10:10:00	1	170	22:45:00		
3044	2017-09-09 11:10:00	2017-09-10 09:55:00	1	97	22:45:00		
196	2017-09-09 18:05:00	2017-09-10 17:10:00	1	170	23:05:00		
3044	2017-09-09 11:10:00	2017-09-10 10:30:00	1	97	23:20:00		
234	2017-09-09 10:25:00	2017-09-10 10:10:00	1	170	23:45:00		
4	2017-09-09 11:25:00	2017-09-10 11:15:00	1	170	23:50:00		

Msg
msg= first flight id:4 first aircraft code:321 scheduled departure:2017-09-09 11:25:00 scheduled arrival:2017-09-09 12:20:00 second flight id:8272 second aircraft code:321 scheduled departure:2017-09-09 16:20:00 scheduled arrival:2017-09-09 17:10:00

En la venta de Msg vemos el flight\_id de primer vuelo: 4, su aircraft\_code llamado first\_aircraft\_code con número 321, luego esta su horario de salida y llegada. En la segunda fila observamos las informaciones de segundo vuelo ya que este resultado es un vuelo con transbordo. Por lo tanto, hay que mostrar las informaciones de los dos vuelos.

### Ejemplo2:

Para este ejemplo hemos introducido From = DME, To = NJC, Date = 2017-09-09

Form Search								
Out								
From:	DME	965	2017-09-09 14:35:00	2017-09-09 17:50:00	0	50	03:15:00	
To:	NJC	1802	2017-09-09 10:15:00	2017-09-09 14:45:00	1	50	04:30:00	
Date:	-09-09	4	2017-09-09 11:25:00	2017-09-09 17:30:00	1	50	06:05:00	
		234	2017-09-09 10:25:00	2017-09-09 17:30:00	1	50	07:05:00	
		76	2017-09-09 08:35:00	2017-09-09 17:30:00	1	50	08:55:00	
		196	2017-09-09 18:05:00	2017-09-10 17:30:00	1	50	23:25:00	

Msg								
From= DME , To= NJC , Date= 2017-09-09								

En la ventana de Out podemos ver los resultados que hemos obtenido. Donde existe vuelos directo y con transbordo entre DME y NJC. En la columna 4 el número 0 significa que es un vuelo directo desde origen hasta destino y el número 1 representa que es un vuelo con transbordo. Si entramos en detalle del vuelo directo de la fila 1:

Menu								
* Search      B. Pass      Quit								
Form Search								
Out								
From:	DME	965	2017-09-09 14:35:00	2017-09-09 17:50:00	0	50	03:15:00	
To:	NJC	1802	2017-09-09 10:15:00	2017-09-09 14:45:00	1	50	04:30:00	
Date:	-09-09	4	2017-09-09 11:25:00	2017-09-09 17:30:00	1	50	06:05:00	
		234	2017-09-09 10:25:00	2017-09-09 17:30:00	1	50	07:05:00	
		76	2017-09-09 08:35:00	2017-09-09 17:30:00	1	50	08:55:00	
		196	2017-09-09 18:05:00	2017-09-10 17:30:00	1	50	23:25:00	

Msg								
msg= first_flight_id:965 first_aircraft_code:CR2 scheduled_departure:2017-09-09 14:35:00 scheduled_arrival:2017-09-09 17:50:00 no hay segundo vuelo								

Observamos que la ventana de Msg aparece las informaciones de primer vuelo con su id, aircraft\_code, horario de salida y horario de llegada. Además, a diferencia de anterior ejemplo, en la fila 2 de Msg muestra el mensaje de no hay segundo vuelos ya que para este resultado concreto es un vuelo directo. En consecuencia, no hay informaciones de segundo vuelo ya que no hace falta.

### Ejemplo 3

En este ejemplo introducimos From = VKO, To = HMA y Date = 2018-08-08

Menu

- \* Search
- B. Pass
- Quit

Form Search	Out
From: VKO To: HMA Date: -08-08	

Msg

No hay vuelo desde VKO a HMA en 2018-08-08

Vemos que la ventana de Out no aparece ningún resultado y si miramos en la venta de Msg observamos que aparece un mensaje que nos informa que no hay vuelo desde VKO a HMA en 2018-08-08

#### Ejemplo 4

Menu

- \* Search
- B. Pass
- Quit

Form Search	Out
From: DME To: _____ Date: -09-09	

Msg

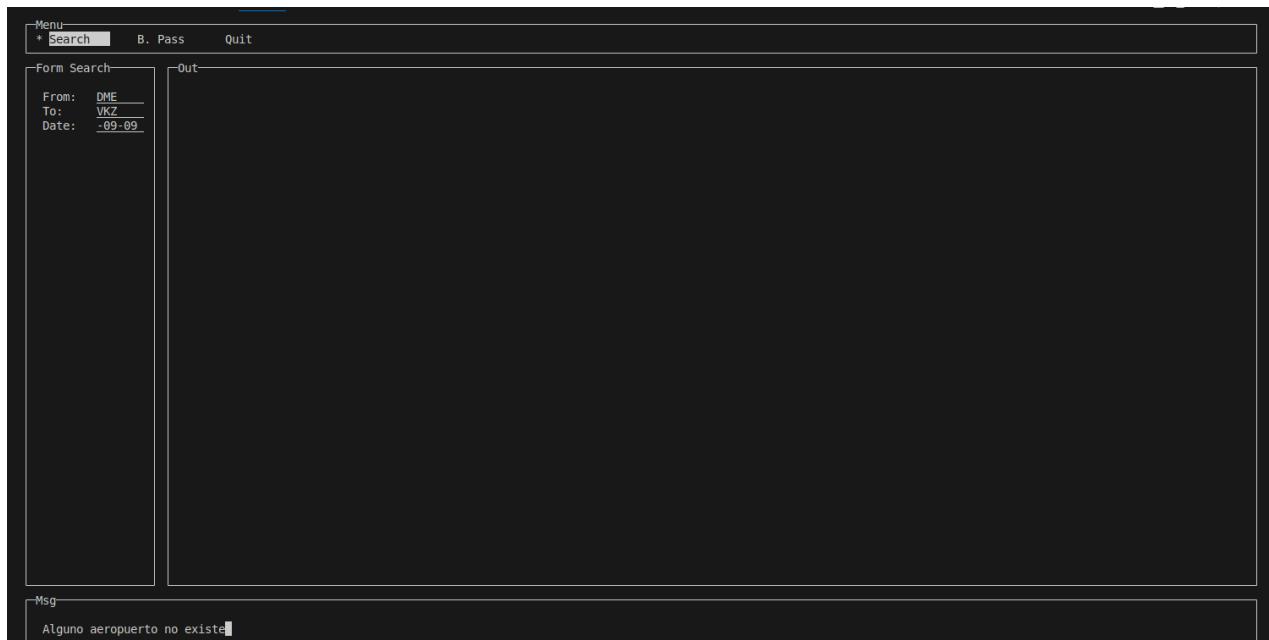
Falta datos para introducir

En este caso vamos a introducir From = DME y Date=2017-09-09 y ponemos To vacío.

Observamos que la ventana de Msg nos informa de que falta datos para introducir

### Ejemplo 5

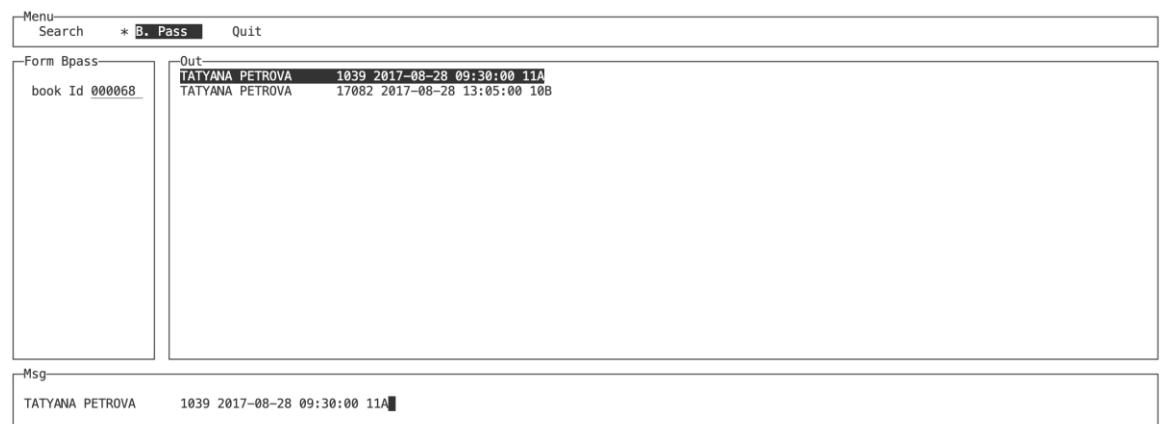
Este ejemplo vamos a comprobar si introducimos un aeropuerto que no existe por ejemplo From = DME y To = VKZ (El aeropuerto que no existe en la base de data flight.sql), Date = 2017-09-09.



Obtenemos un mensaje informativo en la ventana de Msg que nos informa alguno de aeropuerto que hemos introducido no existe.

### b. EMISIÓN DE TARJETAS DE EMBARQUE

Ejemplo 1: book\_ref=000068



En este caso, se ha emitido correctamente 2 tarjetas de embarque para pasajero Tatyana Petrova con book\_ref=000068, uno para flight\_id=1039 y otro para flight\_id=17082 con sus correspondientes asientos asignados y hora de salida de cada vuelo. Así

observamos que

Ejemplo2: book\_ref=000181

Menu	
Search * B. Pass Quit	
Form Bpass	Out
book Id 000181	ALEKSANDR ZHUKOV 1039 2017-08-28 09:30:00 11B ALEKSANDR ZHUKOV 20881 2017-09-08 13:45:00 10A ALEKSANDR ZHUKOV 17082 2017-08-28 13:05:00 10C ALEKSANDR ZHUKOV 11238 2017-09-07 10:40:00 10A EVGENIYA KARPOVA 20881 2017-09-08 13:45:00 10C EVGENIYA KARPOVA 17082 2017-08-28 13:05:00 10D EVGENIYA KARPOVA 1039 2017-08-28 09:30:00 11D EVGENIYA KARPOVA 11238 2017-09-07 10:40:00 10C
Msg	
key pressed 260 ^D (2)	

Como el caso anterior, se ha emitido tarjetas de embarques con los asientos asignados para los pasajeros que todavía no lo tienen, en este caso a varios pasajeros: Aleksandr Zhukov y Evgeniya Karpova, y 4 vuelos para cada pasajero.

Ejemplo 3: book\_ref=00000F

Menu	
Search * B. Pass Quit	
Form Bpass	Out
book Id 00000F	Ya se ha asignado tarjeta de embarque para book_ref=00000F
Msg	
Ya se ha asignado tarjeta de embarque para book_ref=00000F ■	

Y luego en este caso book\_ref=00000F ya tienen los asientos asignados, nos muestra un mensaje “Ya se ha asignado tarjeta de embarque para book\_ref=00000F”.

#### Ejemplo 4: book\_ref=000000

Menu	
Search	* B. Pass
Quit	
Form Bpass	
book Id <u>000000</u>	Out El book_ref=000000 no existe
Msg El book_ref=000000 no existe	

Luego, en este caso probamos con un book\_ref=000000 en el que no existe, y, en definitiva, nos muestra un mensaje “El book\_ref=000000 no existe” indicando que no está registrado por el base de datos.

#### Ejemplo 5: book\_ref=19AFAC

Menu	
Search	* B. Pass
Quit	
Form Bpass	
book Id <u>19AFAC</u>	Out LYUBOV LOGINOVA 14575 2017-09-04 17:55:00 11A LYUBOV LOGINOVA 3505 2017-08-25 10:00:00 11A LYUBOV LOGINOVA 11513 2017-08-26 10:30:00 10A LYUBOV LOGINOVA 10968 2017-09-04 07:50:00 10A LYUBOV LOGINOVA 14596 2017-08-25 14:30:00 10A LYUBOV LOGINOVA 31869 2017-09-03 13:45:00 10A SERGEY ILIN 10969 2017-09-04 07:50:00 10C SERGEY ILIN 11513 2017-08-26 10:30:00 10C SERGEY ILIN 3505 2017-08-25 10:00:00 11C SERGEY ILIN 31869 2017-09-03 13:45:00 10C SERGEY ILIN 14575 2017-09-04 17:55:00 11C
Msg LYUBOV LOGINOVA 14575 2017-09-04 17:55:00 11A	

Menu					
Search * B. Pass Quit					
Form Bpass					
book Id <u>19AFAC</u>					Out
SERGEY ILIN 14696 2017-08-25 14:30:00 10C BOGDAN DMITRIEV 14696 2017-08-25 14:30:00 10D BOGDAN DMITRIEV 18968 2017-09-04 07:50:00 10D BOGDAN DMITRIEV 11513 2017-08-26 10:30:00 10D BOGDAN DMITRIEV 31869 2017-09-03 13:45:00 10D BOGDAN DMITRIEV 3505 2017-08-25 10:00:00 10D BOGDAN DMITRIEV 14575 2017-09-04 17:55:00 11D SERGEY SIDOROV 3505 2017-08-25 10:00:00 11E SERGEY SIDOROV 18968 2017-09-04 07:50:00 10E SERGEY SIDOROV 31869 2017-09-03 13:45:00 10E SERGEY SIDOROV 14575 2017-09-04 17:55:00 11E					
Msg					
key pressed 338 R (1)					

Menu					
Search * B. Pass Quit					
Form Bpass					
book Id <u>19AFAC</u>					Out
SERGEY SIDOROV 14696 2017-08-25 14:30:00 10E SERGEY SIDOROV 11513 2017-08-26 10:30:00 10E					
Msg					
key pressed 338 R (1)					

En este caso para book\_ref=19AFAC se emite 24 tarjetas de embarque para varios pasajeros, y al tener tantas tarjetas emitidas es un buen ejemplo para probar la paginación, así, los resultados se muestran en 3 páginas diferentes, al sumar todas las tarjetas nos den 24.