Tienes razón, hay un problema potencial en tu función buscarReserva. El problema radica en cómo estás usando strcmp y el indicador encontrado dentro del bucle while.

Aquí te explico por qué podría no estar funcionando como se espera y cómo solucionarlo:

**El problema:**

La condición del bucle while (fread(&res, sizeof(Reserva), 1, arch) == 1 && !encontrado) continúa leyendo y sobrescribiendo res incluso si se encuentra una coincidencia. Si la reserva coincidente no es la *última* reserva en el archivo, res será sobrescrita por lecturas posteriores, y la función devolverá el último registro leído, no el que realmente coincidió.

**La solución:**

Necesitas salir del bucle inmediatamente una vez que se encuentra una coincidencia.

Aquí tienes la función buscarReserva corregida:

C++

int buscarReserva(char numero[20], Reserva &res) {  
 int encontrado = 0;  
 FILE \*arch = fopen("CRUCEROS.dat", "rb");  
 if (arch != NULL) {  
 // Crea un objeto Reserva temporal para leer  
 Reserva temp\_res;  
 while (fread(&temp\_res, sizeof(Reserva), 1, arch) == 1) {  
 if (strcmp(temp\_res.numero\_reserva, numero) == 0) {  
 res = temp\_res; // Copia la reserva encontrada a la referencia pasada  
 encontrado = 1;  
 break; // Sale del bucle inmediatamente una vez encontrada  
 }  
 }  
 fclose(arch);  
 }  
 return encontrado;  
}

**Explicación de los cambios:**

1. **Reserva temp\_res;**: He introducido un objeto Reserva temporal llamado temp\_res. Esto es crucial porque fread leerá los datos en temp\_res, dejando res (el objeto Reserva pasado por referencia) intacto hasta que se encuentre una coincidencia.
2. **fread(&temp\_res, sizeof(Reserva), 1, arch) == 1**: Ahora fread lee en temp\_res.
3. **if (strcmp(temp\_res.numero\_reserva, numero) == 0)**: La comparación se realiza con los datos que se acaban de leer en temp\_res.
4. **res = temp\_res;**: Si se encuentra una coincidencia, el contenido de temp\_res (que ahora contiene los datos de la reserva coincidente) se copia a res, el objeto pasado por referencia a la función. Esto asegura que la función main (o dondequiera que se llame a buscarReserva) reciba los datos correctos de la reserva.
5. **break;**: Este es el cambio más importante. Una vez que encontrado se establece en 1 (lo que significa que se ha encontrado una coincidencia), break sale inmediatamente del bucle while. Esto evita que se sigan leyendo y sobrescribiendo res.

Con esta corrección, tu función buscarReserva debería identificar y devolver correctamente la reserva coincidente.