

UD5 – Caso Práctico 1: App de Contactos

OBJETIVO

Implementar un programa modular que sirva de aplicación de contactos.

REQUISITOS

- Se deben poder almacenar hasta un máximo de 100 contactos. Al iniciar la aplicación esta no tendrá ningún contacto almacenado.
- De cada contacto necesitamos saber el nombre, un teléfono y un correo electrónico.
- El usuario interactuará con la aplicación mediante un menú de texto que le permita elegir entre distintas opciones numeradas. Tras cada operación volverá a mostrarse el menú, a no ser que el usuario decida salir de la aplicación. Las opciones son:
 1. Ver contactos.
 2. Agregar contacto.
 3. Eliminar contacto.
 4. Buscar por nombre.
 5. Buscar por teléfono.
 6. Buscar por correo.
 7. Búsqueda global.
 8. Salir.
- **Opción 1. Ver contactos:** Mostrará por pantalla la lista actual de contactos y toda su información, un contacto por línea, y debe estar numerada (ver ejemplo abajo). Si no hay contactos mostrará el mensaje “No hay contactos”.
 - 0. Nombre: María Telf: 600000000 Mail: maria@protonmail.com
 - 1. Nombre: Pepito Telf: 611111111 Mail: pepito@gmail.com
 - ...
- **Opción 2. Agregar contacto:** Permitirá añadir un contacto a la lista de contactos siempre y cuando no esté llena. Se pedirá la información necesaria y se registrará al final de la lista de contactos. No se permite tener dos contactos con el mismo nombre.
- **Opción 3. Eliminar contacto:** Permitirá eliminar un contacto de la lista. Se preguntará qué contacto se desea eliminar y se borrará de la lista.
- **Opciones 4, 5 y 6. Buscar por:** Permitirán buscar un contacto por nombre, teléfono o correo (según la opción elegida). Pedirá al usuario una cadena de texto T y luego mostrará, en forma de lista, la información de todos los contactos que contengan T en su nombre, teléfono o correo (según la opción elegida). No se diferenciará entre mayúscula y minúscula. Si no se encontrara ningún contacto se mostrará el texto “No se han encontrado contactos”.
- **Opción 7. Búsqueda global:** Equivalente a las opciones anteriores con la diferencia de que se mostrará la información de todos los contactos que contengan la cadena de texto T en cualquiera de sus tres campos (nombre, teléfono o correo).
- **Opción 8. Salir:** Terminará el programa.

PASOS

1. **Realiza el diseño descendente (top-down)** del programa, dividiendo el problema en subproblemas más pequeños, sucesivamente hasta que sean simples y no sea necesario dividirlos más.
2. **Piensa de qué forma puedes almacenar la información más importante.** ¿Qué variables o estructuras de datos puedes necesitar? ¿De qué tipo serían?
3. **Identifica qué subproblemas convendría programar como una función.** Identifica también si hay subproblemas parecidos que puedan ser resueltos por la misma función mediante parámetros, o si conviene crear funciones sobrecargadas.
4. **Haz una lista con las funciones** que necesitarás programar (no todos los subproblemas necesitan ser funciones). Incluye la cabecera completa (tipo devuelto, nombre y parámetros) y una comentario explicando qué hace cada una de ellas.
5. **Implementa el programa:** primero todas las funciones y por último el main.