MODUL 14: Tugas Akhir

Pada tugas akhir ini anda diminta untuk membuat sebuah project pemrograman. Pada project ini, anda diminta untuk menerapkan penggunaan struktur data dengan mengimplementasikan sebuah *contact book* atau buku kontak.

Anda boleh mendesain sendiri struktur data yang anda anggap baik, atau anda boleh menggunakan struktur data yang telah anda pelajari sebelumnya. Semua struktur data pada program anda harus didesain dan diimplementasikan sendiri. Jadi anda tidak diperbolehkan menggunakan library tertentu untuk struktur datanya. Pastikan source code dapat dikompilasi tanpa error.

***Contact Book***

Sebuah *contact book* berisi data-data yang anda butuhkan ketika akan menghubungi sebuah teman. Data tersebut dapat berisi bermacam-macam atribut seperti nama, umur, jenis kelamin, nomor telepon, email, tanggal lahir, dan lain-lain. Lihat tabel 1 untuk contoh isi *contact book*. Tantangan utama dari tugas akhir ini adalah anda mampu menggunakan struktur data yang tepat dan efisien yang dapat digunakan untuk mendukung berbagai fungsi atau fitur sebuah *contact book*.

Tabel 1. Contoh struktur *contact book*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nama** | **Umur** | **Jenis Kelamin** | **Nomor Telepon** | **Email** | **…** |
| Andi Pawanari | 17 | L | 081341780123 | [andip@gmail.com](mailto:andip@gmail.com) | … |
| Bebi Putri | 16 | P | 085678912345 | [bebputri@yahoo.com](mailto:bebputri@yahoo.com) | … |
| Catur | 15 | L | 085831124358 | [caturdoang@gmail.com](mailto:caturdoang@gmail.com) | … |
| … | … | … | … | … | … |

**Fitur *contact book***

Fitur-fitur pada *contact book* pada project ini dibagi menjadi 2 kategori yaitu fungsi dasar dan fungsi opsional. Fungsi dasar adalah fitur yang wajib anda bangun sedangkan fitur opsional dapat anda kembangkan untuk menambah nilai akhir.

Berikut ini adalah deskripsi fungsi dasar:

1. ***User Interface***: menampilkan menu utama yang dapat anda pilih dan daftar kontak
2. ***Search***: Pencarian kontak dengan atribut tertentu, menampilkan seluruh atribut lain jika ditemukan
3. ***Insert***: Menambah kontak baru pada buku kontak
4. ***Delete*:** Menghapus kontak dengan atribut tertentu
5. ***Edit*:** Mengupdate atribut kontak tertentu
6. ***Sorting*:** Melakukan pengurutan kontak berdasarkan atribut tertentu. Fungsi ini dapat dijalankan pada semua atribut sesuai kemauan pengguna.

Dan berikut ini adalah deskripsi fungsi opsional

1. ***Save*:** menyimpan data *contact book* yang ada di memori ke dalam sebuah file tertentu. Anda dapat menyimpan data kontak dengan format file anda sendiri.
2. ***Load*:** Membaca data kontak dari file yang telah tersimpan dengan fungsi *save*. Format file harus konsisten dengan fungsi *save.*
3. ***Max* (atau *Min*):** menemukan atribut maksimum atau minimum dari atribut tertentu
4. ***Average*:** menemukan rata-rata dari atribut tertentu yang bertipe numerik
5. ***Undo*:** melakukan undo sebuah action, baik itu undo insert, delete, atau edit
6. ***Redo*:** melakukan redo action yang di-undo sebelumnya
7. ***And*:** Pencarian dengan dua atribut atau lebih, temukan kontak yang match dengan semua kriteria
8. ***Or*:** Pencarian dengan dua atribut atau lebih, temukan kontak yang match dengan paling sedikit satu kriteria
9. ***Wildcard search*:** pencarian dengan wildcard (\*). Misal jika yang dicari nomor telepon 0813\* maka akan ditampilkan daftar kontak dengan nomor awal kontak 0813. Fungsi ini dapat diterapkan pada atribut bertipe string.
10. ***Konektivitas*:** Misal hubungan antar kontak juga tersimpan dalam *contact book*. Relasi ini dapat disimpan di struktur data lain/tambahan. Misal dapat dilihat pada ilustrasi tabel 2 berikut ini:

Tabel 2. Relasi antar kontak dalam *contact book*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Kontak** | **Anda** | **A** | **B** | **C** | **…** |
| **Anda** | 1 | 1 | 1 | 1 | … |
| **A** | 1 | 1 | 1 | 0 | … |
| **B** | 1 | 1 | 1 | 0 | … |
| **C** | 1 | 0 | 0 | 1 | … |
| **…** | … | … | … | … | … |

Misal angka 1 melambangkan memiliki hubungan dan 0 adalah tidak. Jika kontak C ingin menghubungi kontak B, karena C tidak memiliki kontak B, maka C harus menghubungi Anda dahulu untuk mendapatkan kontak B. Contoh lainnya B ingin menghubungi kontak C, namun karena B tidak memiliki kontak C, maka B bisa menghubungi A atau menghubungi Anda dahulu untuk mendapatkan kontak C.

Fungsi ini mengambil input 2 buah kontak, kemudian menentukan apakah kontak pertama dapat menghubungi kontak kedua tersebut secara langsung, jika ya maka selesai. Jika tidak bisa maka melalui kontak mana saja kontak lain tersebut dapat dihubungi.

**Penilaian**

Grade penilaian tugas akhir ditunjukkan pada tabel 3.

Tabel 3. Penilaian tugas akhir

|  |  |  |
| --- | --- | --- |
| **Kategori** | **Fitur** | **Grade** |
| Fungsi Dasar | User Interface | 10% |
| Search | 60% |
| Insert |
| Delete |
| Edit |
| Sorting |
| Fungsi Opsional | Save | 5% |
| Load | 5% |
| Max Min | 3% |
| Average | 3% |
| Undo | 5% |
| Redo | 5% |
| And | 5% |
| Or | 5% |
| Wildcard | 10% |
| Konektivitas | 20% |
| Lain-lain |  | 1-20% |