# KUIS PRA-RESPONSI

**PRAKTIKUM KONSEP PEMROGRAMAN 2021 S1 INFORMATIKA UNS**

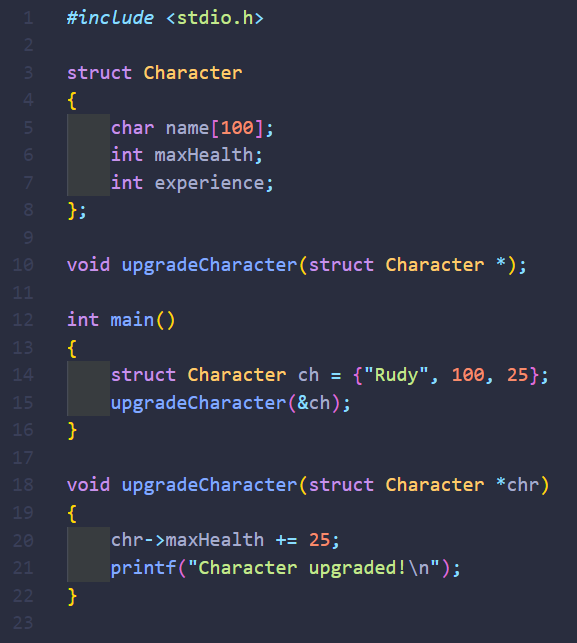
Judul Tugas : Kuis Pra-Responsi

Kelas Praktikum : B – S1 Informatika UNS 2021

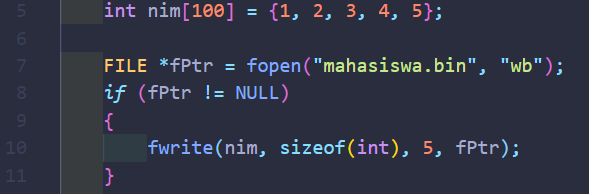
|  |  |
| --- | --- |
| **NO** | **SOAL** |
| 1 | Perhatikan kode di bawah:    Potongan kode di atas apabila di *compile* akan memproduksi suatu eror. Berilah penjelasan singkat mengapa eror tersebut muncul dan tindakan yang harus dilakukan supaya eror  tersebut hilang dan berikan *screenshot* kode pembetulan anda! |
| 2 | Terdapat *array* yang berkapasitas 100 elemen berjenis **int** dan terisi hanya 5 elemen saja. Bagaimana strategi anda dalam menyimpan *array* tersebut ke suatu *binary file*? Jelaskan  secara garis besar lalu berikan potongan kode anda! |
| 3 | Jika anda harus menyimpan data berupa nama dan nomor absen siswa, struktur data apa yang akan anda gunakan untuk mempermudah pencarian berdasarkan nomor absen, dan  mengapa? Jelaskan menurut pendapat anda! |

**Jawab :**

1. Potongan kode pada nomor 1 jika di *compile* akan memproduksi sebuah error pada line ke-17 dikarenakan pada line melakukan perubahan terhadap struct yang telah dideklarasikan menjadi *read-only* (baca-saja) atau *constant struct*. Tindakan yang seharusnya jika memang kita ingin menjumlahkan member *maxHealth* yang berada di dalam struct *Character* maka kita seharusnya tidak mendeklarasikannya menjadi const struct. Sehingga, potongan kode tersebut seharusnya :



1. Karena jumlah elemen array yang akan disimpan ke dalam *binary file* sudah *fixed* (tetap) hanya 5 elemen saja, maka *elementCount* dalam sebuah perintah *fwrite* bisa langsung diisikan dengan 5. Sehingga, potongan kodenya menjadi :



1. Struktur data yang akan digunakan agar mempermudah pencarian berdasarkan nomor absen yaitu *Binary Search Tree (BST)*. Karena untuk melakukan pencarian bisa dengan mudah dilakukan di BST, yaitu dengan mencocokkan *root* terlebih dahulu jika nomor absen yang ingin dicari kurang dari *root* maka pointer bergerak ke kiri, jika data yang ingin dicari lebih dari *root* maka pointer bergerak ke kanan, dan jika data yang ingin dicari sama dengan *root* maka *root* itulah nomor absen yang dicari. Begitupun seterusnya sampai data yang dicari sama dengan *root*/*child* dari BST.