

Nama : Rika Ajeng Finatih

NIM : 121450036

Kelas : RA

Tugas 1

1. Jelaskan menggunakan tabel mengenai kelebihan dan kekurangan paradigma *procedural* dibandingkan dengan *objek oriented* (minimal 3)!

No.	Paradigma Procedural	Objek Oriented Programmer (OOP)
1.	Program utama dibagi menjadi bagian-bagian kecil berdasarkan fungsi dan diperlakukan sebagai program terpisah untuk program yang lebih kecil individu.	Program dibagi menjadi potongan-potongan kecil yang disebut objek yang merupakan turunan dari kelas.
2.	Tidak memiliki access modifiers seperti pada OOP	Memiliki access modifiers seperti public, protected, dan private.
3.	Prosedural kurang aman dibandingkan dengan OOP	Dapat melakukan abstraction dalam penyembunyian data. Oleh karena itu OOP lebih aman daripada Prosedural.
4.	Tidak ada proses sederhana untuk menambahkan data di prosedural, tanpa merevisi seluruh program.	Kurang kompleks dan karenanya objek data baru dapat dibuat dengan mudah dari objek yang ada membuat OOP mudah dimodifikasi.
5.	Cocok untuk project atau aplikasi yang sederhana.	Cocok untuk project atau aplikasi yang kompleks.
6.	Mudah dipahami oleh programmer pemula.	Sedikit sulit dipahami oleh programmer pemula.

2. Jelaskan menggunakan tabel mengenai kelebihan dan kekurangan paradigma *object oriented* dibandingkan dengan *funksional* (minimal 3)!

No.	Objek Oriented	Fungsional
1.	Struktur program ringkas, cukup dengan membuat object dan class, lalu bekerja berdasarkan object dan class tersebut.	Struktur program rumit, karena berupa urutan proses dan fungsi-fungsi.
2.	Kode program sangat reusable. Object dan class dapat digunakan	Re-use kode program kurang.

	berkali-kali, sehingga dapat menghemat space memori.	
3.	Sulit di mengerti bagi programmer pemula karena harus membuat class terlebih dahulu, setelahnya akan terasa mudah jika sudah terbiasa.	Mudah dipahami bagi programmer pemula.

3. Jelaskan menggunakan tabel mengenai kelebihan dan kekurangan paradigma *procedural* dibandingkan dengan *functional* (minimal 3)!

No.	Procedural	Fungsional
1.	Kurang mudah dipahami oleh programmer pemula, karena kode yang ditulis harus bersesuaian dengan step-by step, hal inilah yang sulit untuk memisahkan bagian-bagian yang berbeda dan menggunakan lagi pada bagian lain dalam program.	Memiliki sintaks yang lebih sederhana dan mudah dipahami oleh programmer pemula.
2.	Sulit untuk menggunakan ulang code yang sudah dibuat sebelumnya.	Memungkinkan pemecahan masalah dengan cara yang lebih terstruktur dan logis, dan membuat proses pengembangan kode lebih mudah.
3.	Memiliki kemampuan untuk memecahkan berbagai jenis masalah dan dapat diterapkan pada berbagai lingkungan dan platform.	Sulit untuk membuat program yang skalabel karena tidak ada cara untuk membuat code yang terbagi menjadi bagian-bagian yang lebih kecil dan mudah dikelola.

4. Apa yang dimaksud dengan pure functional programming language?
Pure functional programming language merupakan bahasa pemrograman murni menggunakan persamaan matematika yang diekspresikan sebagai persamaan antar objek matematika dan bukan sebagai persamaan antar variabel. Kemurnian bahasa pemrograman memungkinkan fokus yang lebih besar pada masalah yang dihadapi tanpa mengorbankan keanggunan atau keterbacaan.
5. Dari Bahasa pemrograman FP di slide sebelumnya (slide 14), manakah yang termasuk pure functional programming language!
Haskell, Scala, Javascript, Python.
6. Membuat program simple password encryption, dengan ketentuan sebagai berikut.
- 1) Input password bertipe string dengan batas Panjang password 100 karater.

- 2) Setiap karakter dari string tersebut diubah menjadi ASCII value
- 3) Kemudian setiap ASCII value tersebut dikembangkan menjadi 3 value dengan dilakukan operasi berikut:
 - Value pertama dibagi dengan integer 26 kemudian dibulatkan dan jumlahkan dengan 80 kemudian diubah Kembali menjadi karakter.
 - Value kedua dihitung sisa bagi dengan 26 kemudian dijumlahkan dengan 80 kemudian diubah Kembali menjadi karakter.
 - Value ketiga adalah karakter “+”, jika value pertama lebih besar dari value kedua, atau value ketiga adalah karakter “-“ jika value pertama tidak lebih besar dari value kedua.
- 4) Deretkan value pertama, kedua, dan ketiga untuk setiap karakter dalam password tersebut sehingga didapatkan password dalam bentuk baru.