MATERI PENGAYAAN UKK KELAS XII RPL

Unit Kerja : 1. Menggunakan Struktur Data

Materi : struktur data

1. Array

tipe data terstruktur yang digunakan untuk menyimpan data bertipe sama dalam satu tempat. Array terdiri dari beberapa variabel dengan tipe data yang sama, di mana masing-masing elemen variabel memiliki nilai indeks

Array terdiri dari tiga hal utama, yaitu elemen, indeks, dan panjang. Elemen adalah item yang disimpan dalam array. Sedangkan, indeks merupakan lokasi yang mengidentifikasi elemen dalam array (dimulai dari 0). Kemudian, panjang menunjukkan jumlah elemen yang dapat disimpan array

Ada beberapa macam array, yaitu:

* **Array satu dimensi**: Jenis array yang paling sering digunakan, terutama untuk tipe data string. Contohnya, untuk membuat daftar orang berdasarkan usia.
* **Array dua dimensi**: Array yang terdiri dari m baris dan n kolom. Contohnya, untuk menampilkan data jumlah siswa dan jumlah bangku di sebuah kelas.
* **Array multi dimensi**: Array dengan dimensi yang banyak, biasanya lebih dari dua. Contohnya, array tiga dimensi yang terdiri dari x baris, y kolom, dan z panjang karakter.
* **Indexed arrays**: Jenis array yang paling basic atau mendasar.
* **Multidimensional arrays**: Array yang memiliki banyak dimensi berbeda.
* **Associative arrays**: Array yang cara kerjanya berbeda dengan indexed array.

1. Lingked list

Linked list adalah struktur data yang terdiri dari kumpulan elemen data yang saling terhubung melalui pointer. Setiap elemen dalam linked list disebut node, dan setiap node terdiri dari dua komponen utama, yaitu data dan pointer.

Linked list memiliki beberapa karakteristik, yaitu:

* Dinamis, artinya elemen linked list dibentuk secara dinamis, yaitu sambil jalan sesuai instruksi.
* Terurut, artinya elemen-elemen dalam linked list memiliki keterurutan tertentu.
* Saling terhubung, artinya setiap elemen dihubungkan dengan elemen lain melalui pointer.

Beberapa jenis linked list, antara lain:

* **Single linked list**

Tipe linked list yang paling sederhana, di mana setiap simpul hanya memiliki satu pointer ke simpul berikutnya.

* **Double linked list**

Elemen-elemen dihubungkan dengan dua pointer dalam satu elemen, sehingga list dapat melintas ke depan maupun ke belakang.

* **Circular list**

Bentuk lain dari linked list yang memberikan fleksibilitas dalam melewatkan elemen.

1. LIFO

LIFO adalah singkatan dari Last In First Out, yang merupakan metode pengelolaan barang di mana barang yang terakhir masuk ke gudang akan keluar terlebih dahulu untuk dijual. Metode ini berlawanan dengan metode FIFO (First In First Out) yang menjual barang terlebih dahulu berdasarkan urutan kedatangannya

1. Stack

Stack atau tumpukan adalah struktur data yang digunakan untuk menyimpan dan mengelola data dengan prinsip Last In, First Out (LIFO). Artinya, data yang terakhir dimasukkan ke dalam stack akan menjadi data pertama yang diambil atau dikeluarkan.

Stack memiliki beberapa karakteristik, yaitu:

* Penambahan dan penghapusan elemen hanya dapat dilakukan pada posisi akhir stack.
* Stack dapat diimplementasikan menggunakan struktur data array atau linked list.
* Operasi utama pada stack meliputi Push, Pop, Peek atau Top, IsEmpty, dan Size.

Stack sering digunakan dalam berbagai konteks pemrograman, seperti: Manajemen memori, Pemanggilan fungsi, Evaluasi ekspresi matematika, Manajemen tumpukan panggilan saat melakukan rekursi.

Contoh stack dalam kehidupan sehari-hari adalah tumpukan piring di sebuah restoran. Piring yang terakhir ditambahkan atau ditaruh di atas adalah piring yang paling mudah dilihat

E. Queue

Queue atau antrian adalah struktur data yang menyusun elemen-elemen data dalam urutan linier dengan prinsip First In, First Out (FIFO). Prinsip ini berarti elemen data yang pertama dimasukkan ke dalam antrean akan menjadi yang pertama pula untuk dikeluarkan.

Dalam queue, penambahan elemen dilakukan pada sisi belakang atau rear, sedangkan penghapusan atau pengambilan elemen dilakukan lewat sisi depan atau front.

Contoh penerapan queue dalam kehidupan sehari-hari adalah sistem antrean pelayanan konsumen, pembayaran, dan sebagainya.

Terdapat beberapa jenis queue, di antaranya:

* **Linear Queue**

Elemen data disusun dalam barisan linear, dan proses penambahan serta penghapusan elemen hanya terjadi pada dua ujung barisan tersebut.

* **Simple Queue**

Mirip dengan jenis linear, namun memiliki ujung-ujung barisan yang terhubung satu sama lain, menciptakan struktur antrian yang berputar.

Materi : Pengembangan Aplikasi

* 1. Agile

Metode agile adalah kumpulan metode pengembangan perangkat lunak yang dilakukan secara berulang dan bertahap. Metode ini populer karena dapat tetap efisien dan fleksibel terhadap perubahan yang terjadi saat pembuatan perangkat lunak.

Beberapa hal yang menjadi karakteristik metode agile adalah:

* Fokus pada perkembangan yang cepat
* Mengurangi overhead proses
* Menghasilkan kode berkualitas tinggi
* Memlibatkan pelanggan secara langsung
* Kolaborasi antar anggota tim secara terstruktur dan terorganisir
* Komunikasi yang baik antara anggota tim

Salah satu keuntungan metode agile adalah kemampuannya untuk mempersingkat waktu launching produk. Dengan pendekatan iteratif dan adaptif, produk dapat dikembangkan dan diperbaiki dalam siklus yang lebih singkat

* 1. Prototipe

Metode prototipe adalah metode pengembangan perangkat lunak yang membuat rancangan awal atau versi awal dari sistem untuk menguji dan mengevaluasi konsep sebelum diproduksi secara massal. Metode ini memungkinkan adanya interaksi antara pengembang dan pengguna untuk mengatasi ketidakserasian antara keduanya.

Berikut ini adalah beberapa tahapan dalam metode prototipe:

* **Interaksi dengan pengguna**

Analisis kebutuhan pengguna dari sistem atau perangkat lunak yang akan dibuat.

* **Membuat prototype**

Membuat prototype berdasarkan kebutuhan pengguna dan sistem yang telah diidentifikasi.

* **Pengujian mock-up**

Mendapatkan tanggapan dan kepuasan dari pihak pengguna atas sistem yang telah dirancang.

Beberapa jenis prototype yang dapat digunakan adalah:

* **Requirement prototype**: Pengembang memberikan definisi fungsi dan prosedur sistem yang akan dirancang.
* **Illustrative**: Menghasilkan contoh laporan dan tampilan layar.
* **Simulated**: Mensimulasikan beberapa alur kerja sistem, tetapi tidak menggunakan data real.
* **Functional**: Mensimulasikan beberapa alur sistem yang sebenarnya dan menggunakan data real.
* **Evolutionary**: Menghasilkan model yang menjadi bagian dari operasional sistem
  1. RAD

Rapid Application Development (RAD) adalah metode pengembangan perangkat lunak yang berorientasi objek dan menekankan pada siklus pengembangan yang singkat. Metode ini bertujuan untuk menghasilkan perangkat lunak berkualitas tinggi dalam waktu yang singkat.

Berikut beberapa hal yang perlu diketahui tentang metode RAD:

* **Tahapan**

Tahapan dalam metode RAD meliputi perencanaan kebutuhan, mendesain sistem, proses pengembangan dan pengumpulan feedback, dan implementasi atau penyelesaian produk.

* **Kelebihan**

Metode RAD memiliki beberapa kelebihan, seperti dapat menggunakan kembali komponen yang ada, integrasi proses yang lebih cepat, dan memperkecil kemungkinan kesalahan.

* **Kekurangan**

Metode RAD memiliki beberapa kekurangan, seperti memerlukan kolaborasi tim yang kuat, hanya cocok diterapkan untuk proyek kecil, dan hanya cocok digunakan untuk aplikasi yang memiliki fokus pada suatu fitur.

* **Keterlibatan pengguna**

Metode RAD melibatkan pengguna secara aktif dalam proses pengembangan dengan memberikan umpan balik yang berkelanjutan.

* 1. Scrum

Metode Scrum adalah kerangka kerja manajemen proyek yang membantu tim bekerja secara efisien. Metode ini digunakan untuk:

* Memecah proyek besar menjadi tugas-tugas kecil yang lebih mudah dikelola
* Mengelola dan mengembangkan produk
* Mengatur secara mandiri dan bekerja menuju tujuan bersama
* Menyelesaikan masalah kompleks dengan biaya yang efektif dan berkelanjutan

Metode Scrum menerapkan prinsip-prinsip Agile, seperti transparansi, refleksi, dan adaptasi. Scrum dijalankan melalui tahapan-tahapan yang berulang (iteratif) yang dikenal dengan sebutan Sprint.

Tahapan-tahapan dalam metode Scrum, yaitu:

1. Product Backlog
2. Sprint Planning
3. Sprint
4. Daily Scrum
5. Sprint Review
6. Retrospective Process

Beberapa hal yang perlu diperhatikan dalam metode Scrum, yaitu:

* Tim Scrum terdiri dari 5 hingga 9 orang yang terampil
* Tim Scrum bekerja sama untuk menciptakan produk dan bertanggung jawab atasnya
* Pada fase awal penerapan Scrum, tim biasanya akan memilih durasi Sprint yang paling ideal menurut mereka, yaitu tiga atau empat minggu

E. Waterfall

Metode waterfall adalah metodologi manajemen proyek atau pengembangan perangkat lunak yang dilakukan secara berurutan, seperti aliran air terjun. Dalam metode ini, setiap fase atau tahapan harus diselesaikan sebelum melanjutkan ke fase berikutnya.

Metode waterfall memiliki beberapa kelebihan, di antaranya: Struktur yang jelas, Manajemen proyek yang terprediksi, Dokumentasi yang komprehensif, Meminimalkan kesalahan karena proses pengembangan model dilakukan fase demi fase.

Namun, metode waterfall juga memiliki beberapa kekurangan, seperti: Kurang fleksibel, Terbatas dalam menangani perubahan.

Tahapan-tahapan dalam metode waterfall, antara lain: Analisis kebutuhan, Perancangan, Implementasi, Pengujian, Pemeliharaan.

Metode waterfall pertama kali diperkenalkan oleh Herbert D. Benington pada tahun 1956

Materi : Tool Pendukung Database

* 1. Cassandra Database

Cassandra Database adalah basis data open-source yang dirancang untuk menyimpan dan mengelola data berkapasitas besar. Cassandra merupakan salah satu sistem basis data NoSQL yang paling populer di dunia.

Cassandra memiliki beberapa fitur yang membuatnya menjadi pilihan yang tepat untuk aplikasi berskala besar, di antaranya:

* **Elastic scalability**

Cassandra memungkinkan penambahan kapasitas penyimpanan sewaktu-waktu.

* **Menyimpan data yang fleksibel**

Cassandra mampu mengakomodasi semua format data, termasuk data struktur secara dinamis.

* **Distribusi data yang mudah**

Cassandra memberikan fleksibilitas dalam hal distribusi data sesuai kebutuhan.

* **Menulis cepat**

Cassandra telah dirancang untuk dapat berjalan pada perangkat keras, sehingga dapat menulis dengan cepat.

Cassandra pertama kali dikembangkan oleh Facebook pada tahun 2008, lalu dirilis sebagai proyek open-source pada tahun yang sama

* 1. MongoDB

MongoDB adalah database NoSQL (Not Only SQL) yang menyimpan data dalam format dokumen, seperti JSON (JavaScript Object Notation). MongoDB merupakan database yang populer digunakan untuk pengembangan website dan aplikasi, serta banyak digunakan oleh perusahaan besar seperti Google, Adobe, dan eBay.

|  |  |
| --- | --- |
|  | **MongoDB** |
| Definisi | Database berbasis dokumen yang menyimpan data dalam format JSON |
| Keunggulan | Fleksibel, cepat, dan dapat menangani data yang bervariasi |
| Fitur | Index, replikasi, sharding |
| Bahasa kueri | Bahasa Kueri MongoDB (MQL) |
| Open-source | Ya, sehingga dapat diunduh dan digunakan secara gratis |

MongoDB memiliki beberapa keunggulan, yaitu:

* **Fleksibel**: Struktur datanya yang berbasis dokumen memungkinkan pengguna untuk menyimpan data dengan cara yang sangat fleksibel.
* **Performa cepat**: MongoDB memiliki performa yang sangat cepat dalam memproses data.
* **Open-source**: MongoDB adalah sistem database open-source yang berarti bahwa pengguna dapat mengunduh dan menggunakannya secara gratis.

MongoDB berbeda dengan database relasional, seperti MySQL, yang menyimpan data menggunakan relasi tabel.

* 1. MySQL

MySQL adalah sistem manajemen basis data (DBMS) relasional yang bersifat open source dan gratis:

* MySQL digunakan untuk membuat tabel untuk menyimpan data terkait.
* MySQL menggunakan bahasa pemrograman Structured Query Language (SQL) untuk mengelola informasi di database.
* MySQL dapat digunakan untuk mengakses, mengubah, dan memanipulasi data dalam skala relasional.
* MySQL mendukung berbagai macam data, seperti integer, timestamp, text, date, character, float, dan lain-lain.

MySQL memiliki beberapa kelebihan, di antaranya:

* Gratis dan dapat digunakan oleh siapa saja
* Memiliki sistem keamanan yang baik
* Dapat digunakan secara bersamaan oleh beberapa user
* Mendukung bahasa pemrograman lain, seperti PHP dan JavaScript
* Tidak membutuhkan RAM yang besar
* Unjuk kerja yang tinggi dalam memproses query sederhana
* Memiliki struktur tabel yang lebih fleksibel

MySQL pertama kali dibuat dan diperkenalkan pada tahun 1995 oleh David Axmark dan Michael Wildenius. Saat ini, pengembangan MySQL berada di bawah naungan Oracle.

* 1. PostgreSQL

PostgreSQL adalah sistem manajemen database relasional (RDBMS) yang bersifat open source. PostgreSQL dikenal karena keandalan, fleksibilitas, dan keamanan yang tinggi.

Berikut beberapa fitur PostgreSQL:

* **Open source**

PostgreSQL dapat digunakan dan dimodifikasi secara bebas.

* **Dukungan bahasa**

PostgreSQL kompatibel dengan berbagai bahasa pemrograman, seperti Python, JavaScript, C/C++, dan Ruby.

* **Pemulihan**

PostgreSQL dapat memulihkan database ke momen tertentu saat menjalankan inisiatif pemulihan data.

* **Keamanan**

PostgreSQL menawarkan fitur keamanan yang kuat, termasuk otentikasi, enkripsi, dan kontrol akses yang ketat.

* **Kompatibilitas cross-platform**

PostgreSQL dapat dijalankan di berbagai sistem operasi, termasuk Linux, Windows, dan macOS.

PostgreSQL dikembangkan oleh Berkeley Computer System Department dan pertama kali dikembangkan pada tahun 1986. PostgreSQL digunakan oleh berbagai organisasi, mulai dari startup kecil hingga perusahaan besar

E. Oracle RDBMS

Oracle RDBMS adalah sistem manajemen basis data relasional (RDBMS) yang dikembangkan oleh Oracle Corporation. Oracle RDBMS dirancang untuk menyimpan, mengelola, dan mengakses data secara efisien dan aman.

Oracle RDBMS memiliki beberapa karakteristik, yaitu:

* Menggunakan SQL (Structured Query Language) sebagai bahasa utama untuk mengakses dan memanipulasi data
* Dapat diterapkan di berbagai platform, seperti Windows, UNIX, Linux, dan macOS
* Kompatibel dengan berbagai sistem operasi
* Mendukung transaksi yang sangat besar dan kompleks
* Memiliki fitur keamanan yang kuat
* Dapat digunakan untuk membangun sistem berskala besar, seperti yang ada pada sistem perbankan

Oracle RDBMS dapat memberikan manfaat bagi perusahaan, seperti:

* Meningkatkan efisiensi operasional
* Mengurangi biaya
* Meningkatkan kinerja bisnis secara keseluruhan
* Meningkatkan kualitas data
* Mengoptimalkan rantai pasok
* Memastikan data bisnis tersimpan secara konsisten dan teratur

Unit Kerja : 2. Menggunakan Spesifikasi Program

Materi : Diagram

* 1. Activity Diagram

Activity diagram atau diagram aktivitas adalah diagram yang menggambarkan alur aktivitas atau proses dalam suatu sistem atau bisnis. Diagram ini digunakan untuk membantu pemahaman dan analisis, serta merancang atau memperbaiki sistem informasi.

Activity diagram memiliki beberapa komponen, yaitu:

* **Initial state**

Komponen yang menggambarkan status awal setiap aktivitas, dilambangkan dengan lingkaran hitam kecil di atas diagram

* **Activity**

Komponen yang menggambarkan aktivitas atau proses yang sedang terjadi, dilambangkan dengan persegi panjang dengan ujung melengkung

* **Decision**

Komponen yang menggambarkan kondisi di mana terdapat lebih dari satu alternatif yang mungkin terjadi, dilambangkan dengan belah ketupat

* **Join**

Komponen yang digunakan untuk menghubungkan lebih dari satu aktivitas secara paralel, dilambangkan dengan garis

* **Swimlanes**

Komponen yang memecah activity diagram menjadi kolom dan baris

* **Status akhir**

Komponen yang menandakan berakhirnya activity diagram, dilambangkan dengan lingkaran putih dengan lingkaran hitam di dalamnya

Activity diagram merupakan salah satu jenis Unified Modelling Language (UML) untuk pengembangan software dan berbagai keperluan lainnya

* 1. Flowchart Diagram

Flowchart adalah diagram atau bagan alur yang menggambarkan alur proses atau prosedur menggunakan simbol-simbol standar. Flowchart dapat digunakan untuk menggambarkan alur kerja program, sistem, atau proses lainnya.

Flowchart dapat membantu dalam:

* Merencanakan, menganalisis, dan memahami langkah-langkah yang diperlukan untuk menyelesaikan suatu tugas atau masalah
* Menentukan alur proses
* Menemukan kesalahan atau kekurangan dalam sistem
* Meningkatkan efisiensi suatu proses
* Menjelaskan bagaimana suatu sistem bekerja pada orang yang tidak familiar dengan sistem tersebut

Flowchart dapat dibuat dengan menggunakan simbol-simbol standar seperti simbol dokumen, simbol alur, simbol proses, simbol output, dan simbol input. Flowchart dapat dibuat secara manual atau menggunakan software khusus

* 1. Class Diagram
* *Class diagram* atau diagram kelas adalah salah satu jenis diagram struktur pada UML yang menggambarkan dengan jelas struktur serta deskripsi *class*, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi.
* Diagram kelas ini sesuai jika diimplementasikan ke proyek yang menggunakan konsep object-oriented karena gambaran dari *class diagram* cukup mudah untuk digunakan
  1. Squensial Diagram

1. Sequence diagram atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu sequence diagaram juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek-objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan.
2. diagram ini terdiri dari dua dimensi lo, yaitu dimensi **vertikal** yang menunjukan waktu dan dimensi **horizontal** yang menunjukan objek-objek. Tiap-tiap objek, termasuk actor, memiliki waktu aktif yang digambarkan dengan kolom vertikal yang disebut dengan **lifeline**. Sementara itu, pesan atau perintah digambarkan sebagai garis panah dari satu lifeline ke lifeline yang lain.

E. Use Case Diagram

Use case diagram adalah teknik yang digunakan untuk menggambarkan hubungan antara sistem dan penggunanya. Use case diagram merupakan bagian dari Unified Modelling Language (UML).

Use case diagram dapat digunakan untuk: Mengetahui fungsi-fungsi yang ada pada sistem, Memetakkan kebutuhan sistem, Merepresentasikan interaksi pengguna terhadap sistem, Mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem.

Komponen-komponen pada use case diagram, yaitu:

* **Actor**: Mempresentasikan seseorang atau sesuatu yang berinteraksi dengan sistem.
* **Use Case**: Gambaran fungsionalitas dari suatu sistem.
* **Association**: Menghubungkan link antar elemen.

Use case diagram sering dikaitkan dengan flowchart karena memiliki tampilan yang mirip. Namun, perbedaan yang paling menonjol adalah bahwa use case diagram digunakan untuk menggambarkan interaksi antara aktor dan sistem, sedangkan flowchart berfungsi untuk memperlihatkan alur data

Materi : Menganalisis Kesalahan

* 1. Run+Bug

*bug*diartikan sebagai sebuah kecacatan, *error*, ataupun kegagalan fungsi yang menyebabkan suatu solusi (aplikasi) tidak dapat berjalan sesuai dengan kebutuhan atau harapan. Bug bisa disebabkan oleh berbagai hal, contohnya seperti*human error*, kegagalan integrasi dengan sumber daya eksternal, tidak memberikan validasi pada data yang berasal dari luar, maupun ketidakcocokan antara desain *software*dan lingkungan penggunaan

***bug* adalah** salah satu masalah pada perangkat lunak yang sering terjadi.

Istilah *bug* mengacu pada sebuah kesalahan, cacat, kegagalan, atau kerusakan dalam sebuah perangkat lunak. Apabila ada *bug*, sistem tidak akan berjalan sesuai dengan harapan.

**Jenis-jenis *Bug***

[*Bug*](https://kumparan.com/topic/bug) pada perangkat lunak [komputer](https://kumparan.com/topic/komputer) dapat berbentuk macam-macam. Berdasarkan karakteristiknya, berikut enam jenis *bug* menurut *Jurnal Ilmiah Teknologi Informasi (JUTI) Volume 11 Nomor 2* yang ditulis Sugiyanto.

**1. *Infinite Loop***

*Loop* bisa diartikan sebagai perulangan. Istilah tersebut sering kali digunakan dalam pemrograman. Penggunaan *loop* yang salah akan menyebabkan sebuah program menjalankan sebuah prosedur tanpa akhir (*infinite*).

Informasi penting disajikan secara **kronologis**

[Lihat Breaking News](https://kumparan.com/breakingnews)

**2. *Arithmatic Overflow* atau *Underflow***

ADVERTISEMENT

*Overflow* akan terjadi jika sebuah nilai hasil perhitungan lebih besar daripada nilai yang dapat ditampung oleh variabel penyimpanan. Sebaliknya, *underflow* akan terjadi jika proses kalkulasi menghasilkan nilai lebih kecil daripada nilai yang dapat ditampung oleh variabel penyimpanan.

**3. *Exceeding Array Bounds***

*Array* merupakan variabel berdimensi yang memiliki indeks. Program dapat menjadi [eror](https://kumparan.com/topic/error) saat mengakses indeks di luar *array* yang ditentukan.

**4. *Access Violation***

*Access violation* terjadi saat sebuah proses mencoba melewati batas yang diinginkan sistem. Misalnya, menulis sebuah nilai pada alamat memori, media, atau segmen yang diproteksi.

**5. *Memory Leak***

*Memory leak* adalah penggunaan memori yang tidak diinginkan. Hal ini dapat terjadi karena program gagal melepaskan memori yang sudah tidak digunakan.

ADVERTISEMENT

**6. *Stack Overflow* atau *Underflow***

*Stack* merupakan struktur data yang menggunakan pendekatan *Last in First Out* (LIFO). Sebuah program dapat ditanamkan logika *stack* untuk suatu tujuan. *Stack overflow* atau *underflow* terjadi bila *stack* melebihi atau berada di bawah nilai yang diizinkan oleh program

B. Bugging

C. Running

D. Executable

E. Compiler

1. Running adalah menjalankan keseluruhan program tanpa membuat file .exe untuk desktop atau .apk untuk android
2. debugging adalah menjalankan program dalam setiap 1 baris code, biasa digunakan untuk mencari error atau bug (kesalahan code, data tidak sesuai, atau sejenisnya)
3. executable adalah proses membuat program menjadi file .exe atau .apk

Unit Kerja : 3. Menerapkan Perintah Eksekusi Bahasa Pemrograman Berbasis Teks, Grafik, dan Multimedia

Materi : Tahapan dalam Rekayasa Perangkat Lunak:

1. Analisis

2. Desain

3. Coding

4. Testing

5. Maintenance

Tahapan dalam rekayasa perangkat lunak meliputi:

* **Analisis kebutuhan**: Mengidentifikasi kebutuhan dan tujuan perangkat lunak
* **Perencanaan**: Perencanaan proyek
* **Desain**: Membuat desain perangkat lunak, termasuk wireframe dan maket antarmuka pengguna
* **Implementasi**: Menerapkan desain pada produk
* **Pengujian**: Menguji perangkat lunak untuk memastikan berfungsi dengan baik
* **Dokumentasi**: Membuat dokumentasi kebutuhan, desain, implementasi, dan pengujian
* **Pemeliharaan dan pembaruan**: Memperbarui dan memelihara perangkat lunak secara berkala