

PROPOSAL ALGORITMA PEMROGRAMAN

SISTEM ANALISIS TREN NILAI MAHASISWA PER SEMESTER

1. Latar Belakang

Nilai mahasiswa dari semester ke semester penting untuk mengetahui perkembangan akademik dan menentukan strategi belajar yang tepat. Namun, belum banyak sistem sederhana yang dapat menganalisis tren nilai secara otomatis. Program ini berfokus pada analisis tren nilai mahasiswa per semester, memberikan solusi untuk memantau perkembangan akademik secara periodik. Topik ini kontekstual karena berkaitan langsung dengan kebutuhan mahasiswa dan dosen dalam evaluasi prestasi, serta pengambilan keputusan untuk bimbingan akademik. Keunikannya ada pada penggunaan metode perulangan, percabangan, Binary Search, dan Selection Sort untuk menentukan tren nilai, menemukan semester dengan nilai tertinggi atau terendah, serta menampilkan hasil yang mudah dipahami.

2. Tujuan

- Memantau perkembangan nilai mahasiswa tiap semester
- Menentukan tren nilai (naik, turun, fluktuatif, atau stagnan)
- Mengetahui semester dengan nilai tertinggi dan terendah

3. Skenario Input dan Output (I/O)

| Proses | Input | Output |
|--|---|--|
| Input mahasiswa | Nama, NIM, jumlah semester, IPK tiap semester | Data mahasiswa tersimpan ke array |
| Tampilkan data | - | Semua data mahasiswa dan IPK tiap semester |
| Analisis tren nilai | - | Tren nilai tiap mahasiswa |
| Cari data mahasiswa | Nama mahasiswa | Data mahasiswa yang dicari dan IPK tiap semester, IPK tertinggi dan terendah beserta semesternya |
| Sorting data mahasiswa berdasarkan rata-rata IPK | - | Menampilkan sorting rata-rata IPK mahasiswa |

Contoh data:

Andi (IPK 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7): representatif naik

Citra (IPK 3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 3.0): edge case stagnan

4. Flowchart dan Pseudocode

Flowchart dan Pseudocode

5. Algoritma

- Percabangan (if-else): menentukan tren IPK, logika jelas dan langsung.
- Perulangan (for/while): input, analisis, tampilkan data, efisien untuk tiap mahasiswa/semester.
- Array 1D: menyimpan data mahasiswa/IPK, mudah diakses dan sederhana.
- Bubble Sort: urutkan rata-rata IPK, implementasi mudah untuk data tidak besar.

- Sequential Search: cari mahasiswa berdasarkan nama, cocok untuk jumlah data terbatas.

6. Rencana Subprogram

- a. inputMahasiswa: data[], jumlahMahasiswa - by reference, untuk menambah data mahasiswa.
- b. tampilkanSemuaData: data[], jumlahMahasiswa - by value, menampilkan semua data mahasiswa dan IPK tiap semester.
- c. hitungTren: mhs - by reference, menentukan tren nilai tiap mahasiswa.
- d. cariIPKTertinggiTerendah: mhs - by value, mencari IPK tertinggi dan terendah serta semesternya.
- e. sequentialSearchNIM: data[], jumlahMahasiswa, nama - by value, mencari mahasiswa berdasarkan NIM.
- f. bubbleSortByRataRata: data[], jumlahMahasiswa - by reference, mengurutkan mahasiswa berdasarkan rata-rata IPK.

Tabel Uji:

| No | Skenario Uji | Input | Output yang diharapkan | Jenis Uji |
|----|--|--|--|----------------------|
| 1 | Input mahasiswa | Nama: Andi, NIM: 24051130021 Semester: 8, IPK: 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7 | Data tersimpan di array | Representatif |
| 2 | Input mahasiswa | Nama: Citra, NIM: 24051130022 Semester: 8, IPK: 3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 3.0 | Data tersimpan di array | Edge case (stagnan) |
| 3 | Tampilkan data | - | Semua data mahasiswa dan IPK ditampilkan | Representatif |
| 4 | Analisis tren nilai | - | Tren nilai tiap mahasiswa: Andi (naik), Citra (stagnan) | Representatif + edge |
| 5 | Cari data mahasiswa | Andi | Nama: Andi NIM: 24051130021 IPK: 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7 Tertinggi 3.7 (Semester 8), Terendah 3.0 (Semester 1) | Representatif + edge |
| 6 | Sorting data mahasiswa berdasarkan rata-rata IPK | - | Urutan: Andi, Citra | Representatif |

Repository:

<https://github.com/helgaprayoga/Sistem-Analisis-Tren-Nilai-Mahasiswa-per-Semester>