|  |  |  |
| --- | --- | --- |
| **Nama & NPM** | **Topik:** | **Tanggal:** |
| **M. ARIQ HAFIDL FATANAH  G1F024014**  **M. HOVID ARMAN**  **G1F024034**  **ARIQ ASHAR SOFYAN**  **G1F024052** | **IF SWITCH** | **26 SEPTEMBER 2024** |

**[No. 1] Identifikasi Masalah:**

**1. Diketahui data penelitian mata kuliah di universitas Bengkulu sebagai berikut :**

Nilai AbjadNilai angka MutuRentang Nilai

A 4 85-100

A- 3,75 80-84

B+ 3,5 75-79

B 3 70-74

B- 2,75 65-69

C+ 2,5 60-64

C 2 1

1.1. Rekomendasikan langkah kerja dan flowchart susunan percabangan sesuai dengan data tersebut.

1.2. Desain susunan kode program untuk memeriksa nilai angka yang dimasukkan oleh pengguna ke dalam nilai abjad tertentu,

**2. Konstruksikan kode program dengan IF yang menghitung nilai IPK anda untuk data mata kuliah di semester 1 (gunakan data pada tabel dibawah), dengan variasi nilai abjad yang diinput setiap anggota kelompok harus berbeda dan bervariasi (dalam range A hingga c)**

**Contoh MK (kalian dapat mengubah kolom nilai sesuai keinginan masing-masing):**

Nama Mk SKSNilaiContoh Hitung

Pengantar Teknologi Informasi dan Komunikasi 2 A- 3.75\*2=7.5

Sistem Digital 3 C+ 2.5\*3=7.5

Komputer dan Pemograman 3 A 4\*3=12

Pengantar Sistem multimedia 2 B+ 3.5\*2=7

(7.5 + 7.5 + 12 +7)/10

IPK 10

IPK = 3.4

Petunjuk:

Rumus perhitungan IPK = (total jumlah (sks nilai)) / jumlah sks

**[No.1] Analisis dan Argumentasi**

**Analisis**

1. Penggunaan Fungsi:

Penggunaan fungsi untuk menentukan nilai abjad dan menghitung IPK sangat penting dalam pemrograman karena meningkatkan keterbacaan dan modularitas. Dengan memisahkan logika menjadi fungsi-fungsi terpisah, kita dapat lebih mudah memahami, memelihara, dan menguji kode.

1. Pengkategorian Nilai:

Proses pengkategorian nilai angka menjadi nilai abjad menggunakan struktur percabangan (if-elif) memungkinkan program untuk secara sistematis mengevaluasi rentang nilai. Pendekatan ini efisien untuk memetakan rentang yang ditentukan.

1. Perhitungan IPK:

Menghitung IPK dengan formula yang merupakan pendekatan yang umum digunakan dalam sistem pendidikan.

1. Validasi Nilai:

Penting untuk menyertakan validasi nilai input dari pengguna untuk memastikan bahwa nilai yang dimasukkan berada dalam rentang yang valid.

**Argumentasi**

1. Kepentingan Sistem Evaluasi yang Transparan:

Sistem evaluasi yang jelas dan transparan, seperti yang disajikan dalam kode, memungkinkan mahasiswa memahami bagaimana nilai mereka dihitung dan apa yang diperlukan untuk mencapai IPK tertentu. Ini dapat mendorong mahasiswa untuk lebih berusaha dalam studi mereka.

1. Fleksibilitas dan Skalabilitas:

Struktur yang dirancang menggunakan fungsi memungkinkan penambahan mata kuliah baru atau perubahan rentang nilai dengan mudah tanpa mempengaruhi bagian lain dari kode. Ini menjadikan program lebih fleksibel dan mudah diperbarui seiring waktu.

1. Pentingnya Analisis Data Akademik:

Program semacam ini bukan hanya alat untuk menghitung IPK, tetapi juga dapat digunakan untuk analisis lebih lanjut, seperti memantau tren prestasi akademik dari waktu ke waktu. Dengan data yang tepat, universitas dapat mengambil keputusan yang lebih baik terkait kurikulum dan dukungan akademik.

1. Keterlibatan Pengguna:

Meminta input dari pengguna meningkatkan keterlibatan dan memastikan bahwa hasil yang diperoleh relevan dengan kebutuhan pengguna. Ini juga memberikan kesempatan bagi mahasiswa untuk aktif dalam proses penilaian akademik mereka.

1. Peningkatan Pengalaman Pengguna:

Dengan menampilkan hasil secara jelas dan terstruktur, pengalaman pengguna akan lebih baik. Ini juga membuka jalan untuk fitur tambahan, seperti menyimpan dan membandingkan hasil dengan semester sebelumnya, yang dapat menjadi alat bantu untuk perencanaan studi ke depan.

**[No.1 ] Penyusunan Algoritma dan Kode Program**

1. Algoritma no.1 1) Mulai.

2) Input nilai pengguna.

3) Pengecekan kondisi input:

- Jika nilai angka antara 85 dan 100, maka, set nilai abjad menjadi "A".

- Jika nilai angka antara 80 dan 84, set nilai abjad menjadi "A-".

- Jika nilai angka antara 75 dan 79, set nilai abjad menjadi "B+".

- Jika nilai angka antara 70 dan 74, set nilai abjad menjadi "B".

- Jika nilai angka antara 65 dan 69, set nilai abjad menjadi "B-".

- Jika nilai angka antara 60 dan 64, set nilai abjad menjadi "C+".

- Jika nilai angka kurang dari 60, set nilai abjad menjadi "C".

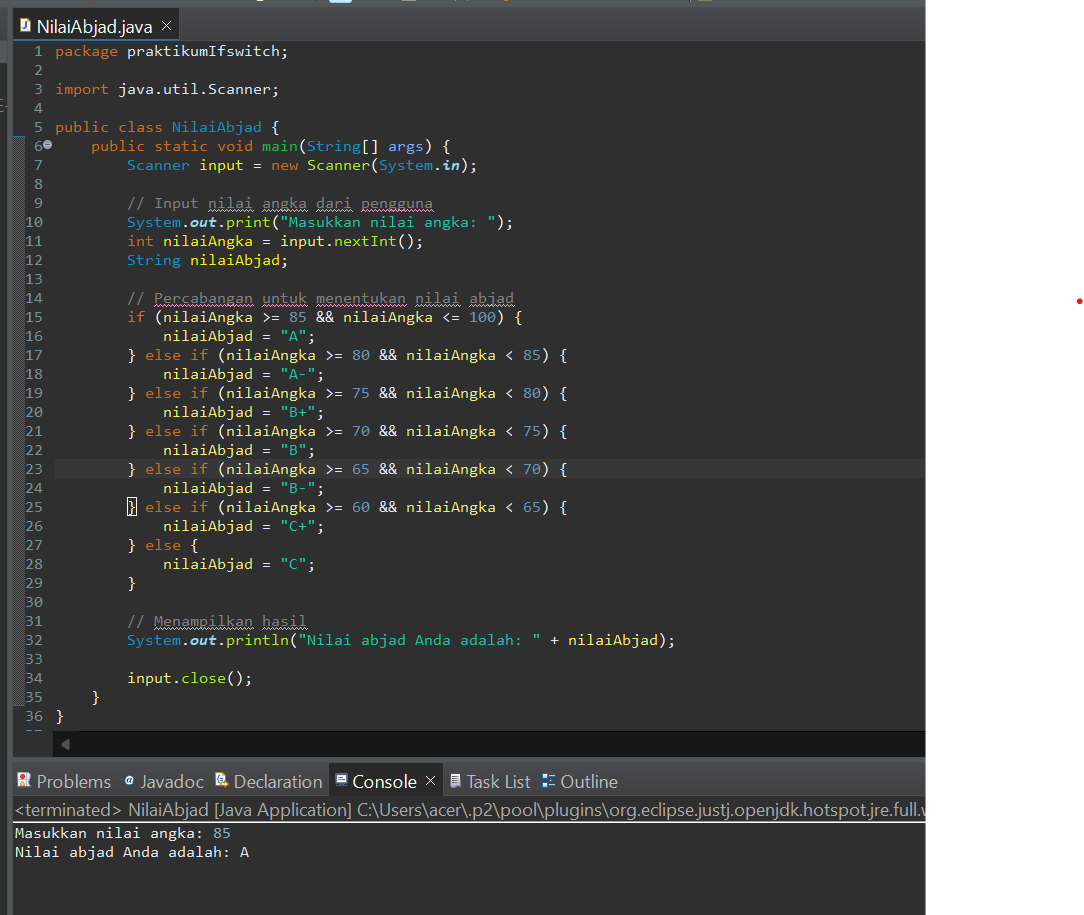
4) Tampilkan nilai abjad.

5) Selesai.

1. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

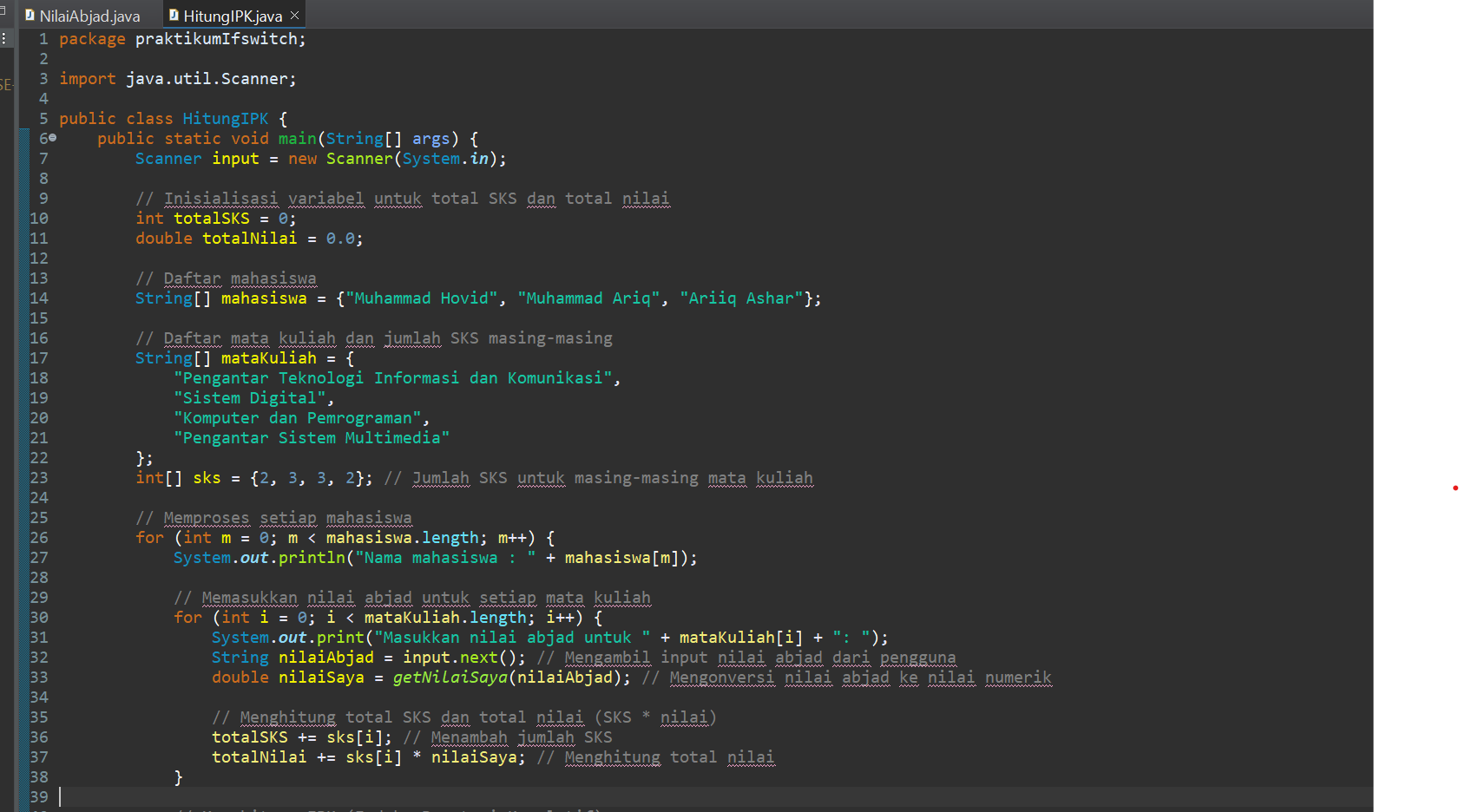
Contoh 1.

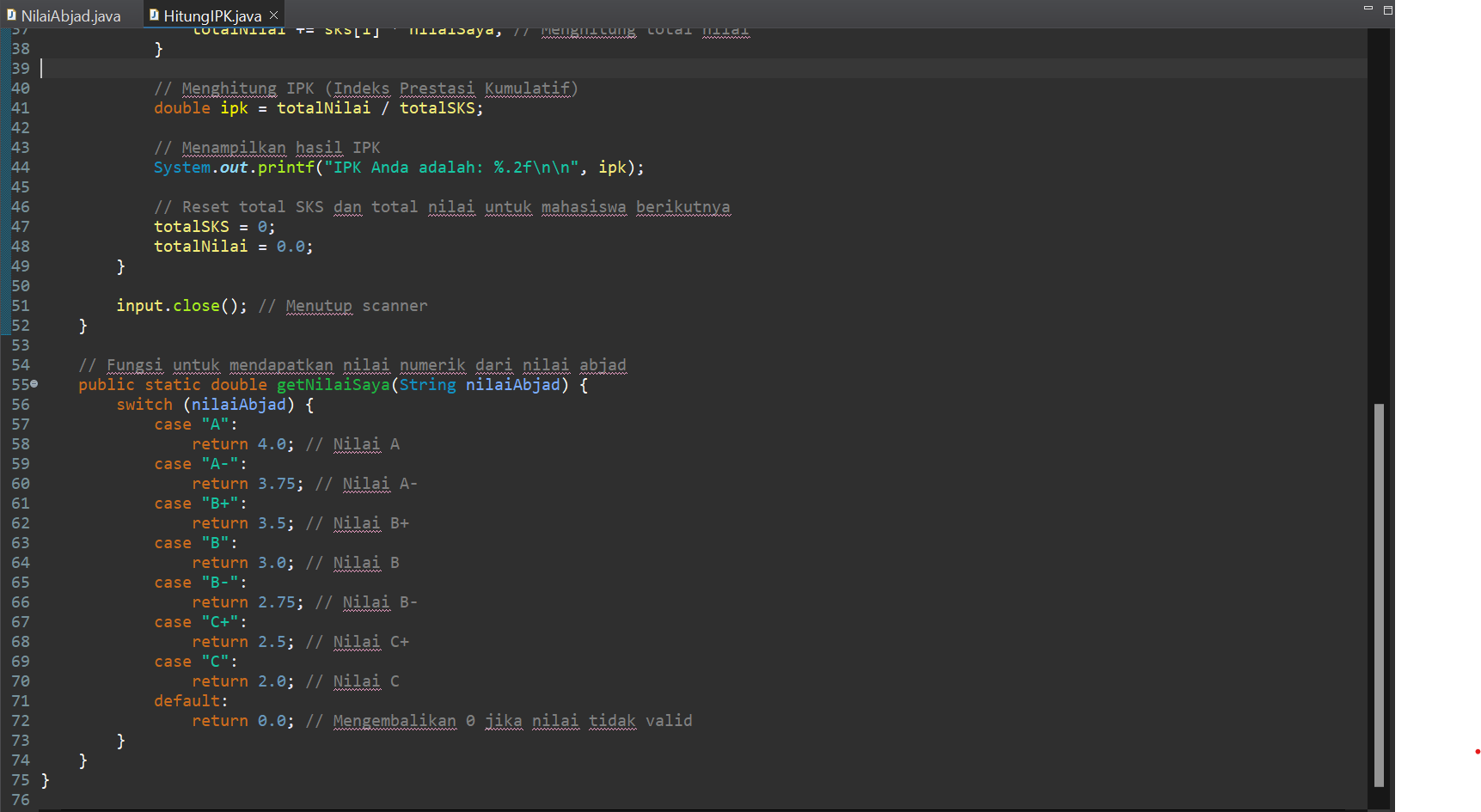


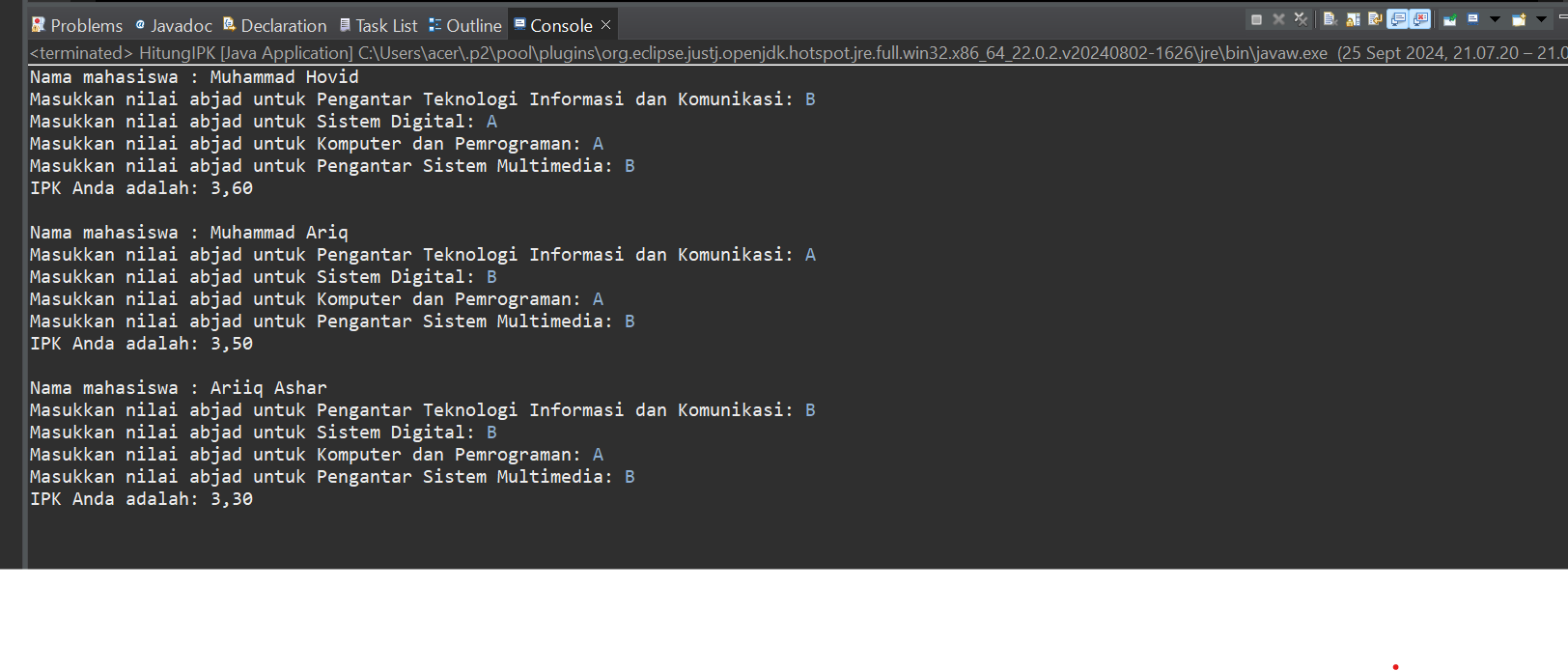
1. Algoritma no.2
2. Mulai
3. Inisialisasi Variabel:
4. Buat variabel totalSKS untuk menyimpan total SKS.
5. Buat variabel totalNilai untuk menyimpan total nilai.
6. Buat array mahasiswa .
7. Buat array mataKuliah.
8. Buat array jumlah SKS untuk masing-masing mata kuliah.
9. Proses untuk Setiap Mahasiswa:
10. Untuk setiap mahasiswa dalam array mahasiswa:
11. Tampilkan nama mahasiswa.
12. Untuk setiap mata kuliah dalam array mataKuliah:
13. Minta pengguna untuk memasukkan nilai abjad untuk mata kuliah tersebut.
14. Konversi nilai abjad ke nilai numerik menggunakan fungsi getNilaiSaya.
15. Tambahkan jumlah SKS ke totalSKS.
16. Hitung total nilai dengan mengalikan SKS mata kuliah dengan nilai numerik dan tambahkan ke totalNilai.
17. Hitung IPK dengan membagi totalNilai dengan totalSKS.
18. Tampilkan hasil IPK dengan format dua desimal.
19. Reset totalSKS dan totalNilai untuk mahasiswa berikutnya.
20. Tutup Scanner
21. Selesai.
22. Screenshot/ Capture potongan kode dan hasil luaran

Beri komentar pada kode yang di Screenshot

Contoh 2.



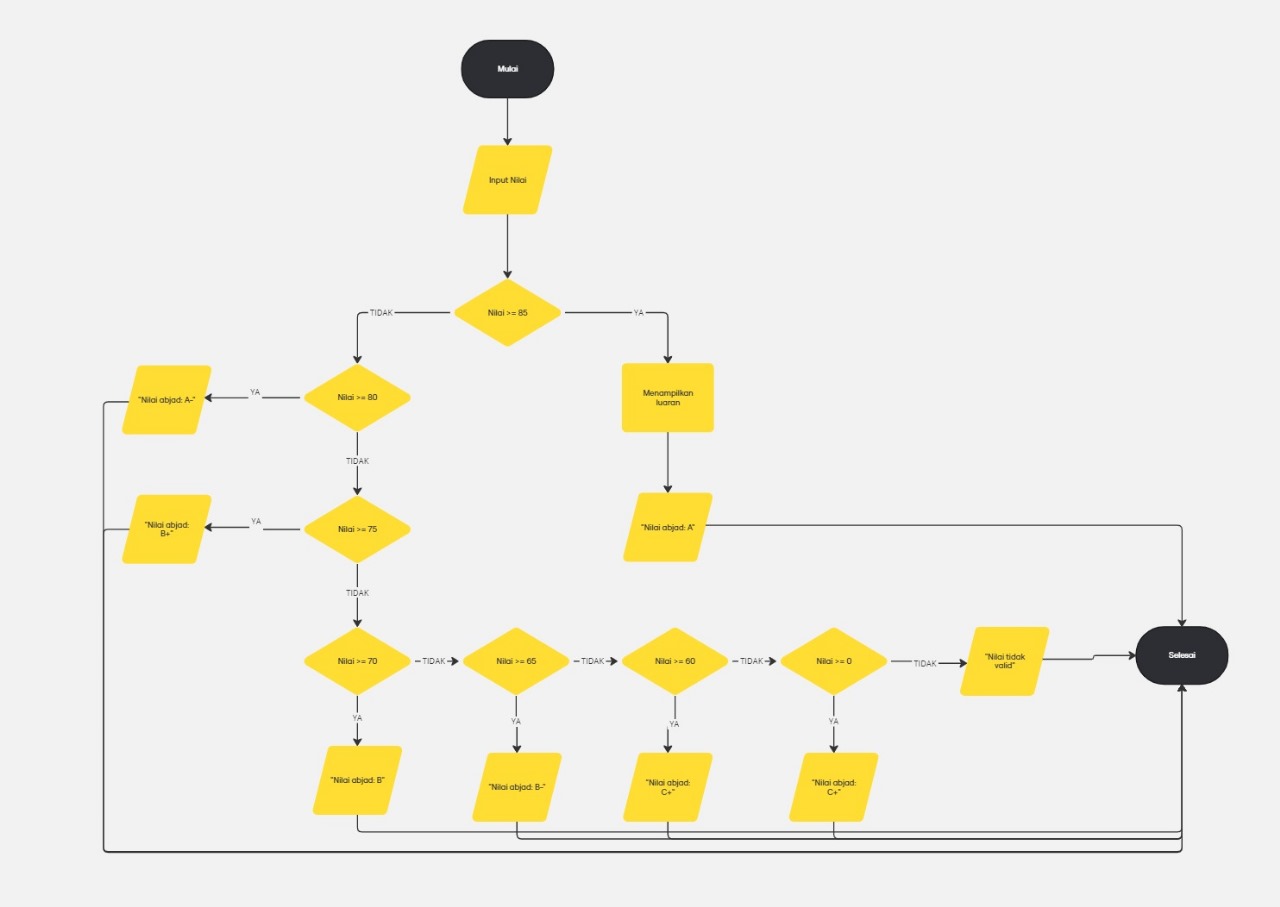




1. Analisa luaran yang dihasilkan

* Contoh 1
  + Luaran yang dihasilkan adalah nilai abjad A karena pengguna menginput nilai 85 sehingga program memproses kondisi if yang pertama dengan rentang nilai 85 sampai 100
* Contoh 2
  + Luaran yang dihasilkan sesuai dengan input data yang dimasukkan pengguna dan diproses oleh kode program yang disusun.

**3) FlowChart**

****

**[No.1] Kesimpulan**

1. **Analisa**

* Program ini dapat digunakan untuk mengonversi nilai angka menjadi nilai abjad dan menghitung IPK berdasarkan mata kuliah yang diambil dan sks yang ada.
* Algoritma yang digunakan sederhana dan efisien, memanfaatkan percabangan untuk menentukan nilai abjad dan melakukan sebuah perhitungan yang matematis untuk IPK.
* Penggunaan percabangan IF dalam kode program didasarkan pada kebutuhan agar dapat melakukan evaluasi rentang nilai yang berbeda dan memberikan output yang sesuai.

**Refleksi**

Materi IF SWITCH ini merupakan materi yang cukup sulit untuk dipahami karena membutuhkan tingkat ketelitian dan fokus yang tinggi. Hal itu dibutuhkan agar dapat mengurangi error pada program, karena IF SWITCH memerlukan penyusunan kode logika yang tepat agar dapat berjalan sesuai dengan yang diinginkan oleh pengguna. Dalam penyusunan switch case, kita tidak boleh lupa menambahkan break setelah membuat kondisi untuk suatu case, agar program dapat menghentikan proses pemeriksaan dan menampilkan hasil untuk kondisi tersebut, sehingga tidak bergabung ke kondisi selanjutnya. Lupa menambahkan break dapat menyebabkan program berjalan tidak sesuai dengan yang diinginkan, mengakibatkan bug atau kesalahan dalam alur logika.

Selain itu, penyusunan kode program ini juga merupakan hal yang sulit, terutama pada bagian pembuatan flowchart. Karena ini merupakan pengalaman pertama dalam membuat flowchart, menentukan bentuk shape yang tepat dan memetakan kode yang akan dimasukkan dalam diagram terasa menantang. Pemahaman terhadap bentuk-bentuk dasar flowchart seperti decision, process, dan input/output.

Meskipun penyusunan tugas kami belum sempurna, kami telah berusaha untuk melakukan yang terbaik. Kami menyadari bahwa masih banyak yang perlu diperbaiki, terutama dalam hal logika kode dan pembuatan flowchart. Namun, pengalaman ini menjadi pembelajaran penting bagi kami untuk lebih memahami materi dan memperbaiki kelemahan dalam pembuatan program di masa mendatang. Dengan lebih banyak latihan dan pemahaman mendalam, kami yakin penyusunan kode logika dan flowchart dapat kami tingkatkan di kesempatan berikutnya.