**UNTUK NOMOR URUT PRESENSI GANJIL**

**Tugas Studi Kasus: Algoritma Aplikasi Kasir Kedai Kopi "Ngopi Dulu"**

**Deskripsi Kasus**

Anda diminta untuk merancang logika (algoritma) untuk aplikasi kasir sederhana di sebuah kedai kopi fiktif bernama "Ngopi Dulu". Aplikasi ini harus dapat menghitung total belanjaan pelanggan, menerapkan diskon, dan menghitung kembalian.

**Ketentuan Bisnis:**

1. **Menu dan Harga:**
   * Kopi Susu: Rp 18.000
   * Americano: Rp 15.000
   * Croissant: Rp 12.000
   * Donat Gula: Rp 8.000
2. **Diskon dan Promosi:**
   * **Diskon Member:** Pelanggan yang merupakan member mendapatkan diskon sebesar **10%** dari total belanja.
   * **Promo Spesial:** Jika total belanja (sebelum diskon) melebihi **Rp 50.000**, pelanggan mendapatkan bonus 1 Donat Gula gratis.

**Tugas Anda**

Buatlah algoritma untuk menyelesaikan kasus di atas. Algoritma tersebut harus disajikan dalam **tiga bentuk**:

1. **Kalimat Deskriptif:** Jelaskan langkah-langkahnya menggunakan bahasa Indonesia yang terstruktur dan mudah dipahami.
2. **Pseudocode:** Tuliskan algoritma menggunakan notasi pseudocode yang lebih formal.
3. **Flowchart:** Gambarkan alur logika algoritma menggunakan simbol-simbol flowchart standar.

**Alur Logika yang Diharapkan**

Aplikasi harus mengikuti alur berikut:

1. Aplikasi memulai sesi transaksi baru.
2. Aplikasi akan **berulang kali** meminta kasir untuk memasukkan item yang dibeli pelanggan (misalnya dengan kode produk 1 untuk Kopi Susu, 2 untuk Americano, dst.) sampai kasir mengindikasikan selesai (misalnya dengan memasukkan kode '0').
3. Setiap item yang dimasukkan akan ditambahkan ke total belanjaan.
4. Setelah selesai memasukkan semua item, aplikasi akan menanyakan **apakah pelanggan adalah seorang member** (input: Ya/Tidak).
5. Aplikasi akan memeriksa **apakah total belanjaan awal melebihi Rp 50.000** untuk menentukan bonus.
6. Aplikasi menghitung total akhir setelah menerapkan diskon (jika ada).
7. Aplikasi menampilkan struk ringkas yang berisi:
   * Subtotal belanja.
   * Diskon yang didapat (jika ada).
   * Bonus yang didapat (jika ada).
   * Total yang harus dibayar.
8. Kasir memasukkan jumlah uang yang dibayarkan pelanggan.
9. Aplikasi menghitung dan menampilkan uang kembalian.
10. Sesi transaksi selesai.

**Identifikasi Elemen Algoritma dalam Kasus Ini**

Untuk membantu Anda, berikut adalah pemetaan elemen algoritma yang harus ada dalam solusi Anda:

* **Urutan (Sequence):**
  + Menghitung subtotal **HARUS** dilakukan sebelum mengecek status member.
  + Menghitung diskon **HARUS** dilakukan sebelum menghitung total akhir.
  + Menerima pembayaran **HARUS** dilakukan sebelum menghitung kembalian.
* **Perulangan (Repetition):**
  + Proses **memasukkan item belanjaan** oleh kasir. Proses ini akan terus diulang sampai kasir menyatakan selesai. Ini adalah kandidat utama untuk loop WHILE atau DO-WHILE.
* **Percabangan (Selection):**
  + **Kondisi 1:** IF pelanggan adalah member, THEN hitung diskon 10%, ELSE tidak ada diskon.
  + **Kondisi 2:** IF subtotal belanja > Rp 50.000, THEN tambahkan bonus "Donat Gula", ELSE tidak ada bonus.
  + **Kondisi 3 (Opsional tapi baik):** IF uang yang dibayarkan cukup, THEN hitung kembalian, ELSE tampilkan pesan "Uang tidak cukup".
  + **Kondisi 4 (di dalam perulangan):** Saat memilih item, program akan memeriksa kode yang dimasukkan (misalnya, IF kode == 1, THEN tambah harga Kopi Susu).

**Contoh Skenario untuk Pengujian Logika**

* **Pelanggan:** Non-Member
* **Belanjaan:** 1 Kopi Susu (Rp 18.000), 2 Croissant (2 x Rp 12.000 = Rp 24.000)
* **Subtotal:** Rp 18.000 + Rp 24.000 = Rp 42.000
* **Apakah Member?** Tidak -> Diskon = 0.
* **Apakah Subtotal > Rp 50.000?** Tidak -> Tidak ada bonus.
* **Total Bayar:** Rp 42.000
* **Uang Pelanggan:** Rp 50.000
* **Kembalian:** Rp 8.000

**Pengumpulan Tugas:** Tugas dikumpulan dalam bentuk paper dalam file \*.pdf, didalam paper tersebut memuat jawaban atau langkah-langka penyelesaian. Dan sertakan file praktikum dengan flowghorithm dan file python

**UNTUK NOMOR URUT PRESENSI GENAP**

**Tugas Studi Kasus: Program Kalkulator Nilai Akhir Mahasiswa**

**Deskripsi Kasus**

Anda adalah seorang asisten dosen yang ditugaskan untuk membuat sebuah program sederhana untuk menghitung nilai akhir mahasiswa dalam satu mata kuliah. Program ini harus bisa menerima input nilai dari beberapa komponen, menghitung nilai akhir berdasarkan bobot tertentu, menentukan Indeks Prestasi (dalam bentuk huruf), dan menyatakan status kelulusan mahasiswa.

Program ini juga harus bisa digunakan untuk menghitung nilai lebih dari satu mahasiswa tanpa harus me-restart program setiap kali.

**Ketentuan Akademik:**

1. **Komponen dan Bobot Nilai:**
   * Absensi: 10%
   * Tugas: 20%
   * UTS (Ujian Tengah Semester): 30%
   * UAS (Ujian Akhir Semester): 40%
   * Rumus Nilai Akhir: (Nilai Absensi \* 0.10) + (Nilai Tugas \* 0.20) + (Nilai UTS \* 0.30) + (Nilai UAS \* 0.40)
2. **Konversi Nilai ke Indeks Huruf (Grading):**
   * > 80 hingga 100 : A
   * > 70 hingga 80 : B
   * > 60 hingga 70 : C
   * > 50 hingga 60 : D
   * <= 50 : E
3. **Syarat Kelulusan:**
   * Mahasiswa dinyatakan LULUS jika Nilai Akhir lebih besar dari 60 (minimal C) DAN Nilai UAS minimal 50. Jika salah satu syarat tidak terpenuhi, mahasiswa dinyatakan TIDAK LULUS.

**Tugas Anda**

Rancanglah algoritma untuk program kalkulator nilai ini. Algoritma tersebut harus disajikan dalam tiga bentuk:

1. Kalimat Deskriptif: Uraikan alur kerja program langkah demi langkah.
2. Pseudocode: Sajikan logika dalam format pseudocode yang terstruktur.
3. Flowchart: Visualisasikan alur algoritma menggunakan bagan alir.

**Alur Logika yang Diharapkan**

Program akan berjalan dalam sebuah siklus (loop) utama:

1. Di awal, program akan bertanya kepada pengguna, "Apakah Anda ingin menghitung nilai mahasiswa? (Y/T)".
2. Selama pengguna menjawab 'Y' (Ya), program akan melakukan langkah-langkah berikut: a. Meminta pengguna memasukkan Nama Mahasiswa. b. Meminta pengguna memasukkan empat nilai komponen: Nilai Absensi, Tugas, UTS, dan UAS. c. Menghitung Nilai Akhir menggunakan rumus bobot yang telah ditentukan. d. Menentukan Indeks Huruf (A/B/C/D/E) berdasarkan rentang Nilai Akhir. e. Menentukan status kelulusan ("LULUS" atau "TIDAK LULUS") berdasarkan dua syarat yang ada. f. Menampilkan semua hasil secara terstruktur: Nama, Nilai Akhir, Indeks Huruf, dan Status Kelulusan. g. Setelah selesai, program kembali bertanya, "Apakah Anda ingin menghitung nilai mahasiswa lain? (Y/T)".
3. Jika pengguna menjawab 'T' (Tidak) atau input lainnya, program akan menampilkan pesan "Terima kasih!" dan berhenti.

**Identifikasi Elemen Algoritma dalam Kasus Ini**

* Urutan (Sequence):
  + Input nilai harus dilakukan sebelum proses perhitungan.
  + Perhitungan Nilai Akhir harus dilakukan sebelum penentuan Indeks Huruf.
  + Penentuan Indeks Huruf dan status kelulusan harus dilakukan sebelum menampilkan hasil akhir.
* Perulangan (Repetition):
  + Keseluruhan proses dari meminta input data mahasiswa hingga menampilkan hasil berada di dalam sebuah *loop*. Program akan terus mengulang proses ini selama pengguna masih ingin memasukkan data mahasiswa baru.
* Percabangan (Selection):
  + Kondisi 1 (Bertingkat): Untuk menentukan Indeks Huruf. Program perlu memeriksa IF Nilai Akhir > 80, ELSE IF Nilai Akhir > 70, ELSE IF Nilai Akhir > 60, dan seterusnya.
  + Kondisi 2 (Logika AND): Untuk menentukan status kelulusan. Program perlu memeriksa IF (Nilai Akhir > 60) AND (Nilai UAS >= 50), THEN LULUS, ELSE TIDAK LULUS.
  + Kondisi 3 (Kontrol Perulangan): WHILE pengguna menjawab 'Y', jalankan program. Ini adalah kondisi yang mengontrol loop utama.

**Contoh Skenario untuk Pengujian Logika**

* Input:
  + Nama Mahasiswa: Citra
  + Nilai Absensi: 100
  + Nilai Tugas: 85
  + Nilai UTS: 70
  + Nilai UAS: 65
* Proses Perhitungan:
  + Nilai Akhir = (100 \* 0.10) + (85 \* 0.20) + (70 \* 0.30) + (65 \* 0.40)
  + Nilai Akhir = 10 + 17 + 21 + 26 = 74
* Proses Penentuan:
  + Indeks Huruf: Karena 74 berada di antara >70 dan 80, maka Indeksnya adalah 'B'.
  + Status Kelulusan: (Nilai Akhir 74 > 60) adalah Benar. (Nilai UAS 65 >= 50) adalah Benar. Karena keduanya benar, statusnya adalah LULUS.
* **Output yang Diharapkan:**
  + **--- Hasil Perhitungan ---**
  + **Nama Mahasiswa: Citra**
  + **Nilai Akhir: 74**
  + **Indeks Huruf: B**
  + **Status: LULUS**