

[T1] Toko Pak Budi

Struktur Data dan Algoritma
Departemen Teknik Komputer ITS

1 Deskripsi Masalah

Pak Budi adalah seorang pedagang di Pasar Keputran yang baru saja memutuskan untuk menggunakan sistem digital dalam mengelola tokonya. Sebagai seorang yang masih awam dengan teknologi, Pak Budi membutuhkan sistem yang sederhana namun efektif untuk mengelola inventori barang dagangannya.

Setiap barang di toko Pak Budi memiliki informasi sebagai berikut:

- **Kode:** Nomor identifikasi unik barang (integer positif)
- **Nama:** Nama barang (string tanpa spasi)
- **Stok:** Jumlah barang yang tersedia (integer non-negatif)
- **Harga:** Harga jual per unit dalam rupiah (integer positif)

Pak Budi ingin sistem yang dapat melakukan operasi berikut:

1. **TAMBAH:** Menambahkan barang baru ke inventori
2. **CARI:** Mencari barang berdasarkan kode barang
3. **JUAL:** Menjual sejumlah barang (mengurangi stok)
4. **RESTOK:** Menambah stok barang yang sudah ada
5. **LAPORAN:** Menampilkan semua barang yang diurutkan berdasarkan kode

Karena Pak Budi masih belajar menggunakan komputer, ia sering melakukan kesalahan input. Oleh karena itu, sistem harus dapat menangani berbagai kondisi error dengan memberikan pesan yang jelas.

2 Format Masukan dan Keluaran

2.1 Format Masukan

Masukan terdiri atas beberapa baris. Baris pertama berisi satu buah bilangan bulat positif N ($1 \leq N \leq 100$) yang merupakan banyak operasi. N baris berikutnya masing-masing berisi operasi yang akan dilakukan dengan format sebagai berikut:

- TAMBAH kode nama stok harga

- CARI kode
- JUAL kode jumlah
- RESTOK kode jumlah
- LAPORAN

Batasan:

- $1 \leq kode \leq 9999$
- $1 \leq |nama| \leq 20$, tanpa spasi
- $0 \leq stok \leq 1000$
- $1 \leq harga \leq 1000000$
- $1 \leq jumlah \leq 100$

2.2 Format Keluaran

Untuk setiap operasi, keluarkan respons sesuai format berikut:

- **TAMBAH:**
 - Jika berhasil: BARANG DITAMBAHKAN
 - Jika kode sudah ada: KODE SUDAH ADA
- **CARI:**
 - Jika ditemukan: kode nama stok harga
 - Jika tidak ditemukan: BARANG TIDAK DITEMUKAN
- **JUAL:**
 - Jika berhasil: BERHASIL TERJUAL
 - Jika barang tidak ada: BARANG TIDAK DITEMUKAN
 - Jika stok tidak cukup: STOK TIDAK CUKUP
- **RESTOK:**
 - Jika berhasil: STOK DITAMBAHKAN
 - Jika barang tidak ada: BARANG TIDAK DITEMUKAN
- **LAPORAN:** Tampilkan semua barang yang diurutkan berdasarkan kode (ascending). Jika tidak ada barang: TIDAK ADA BARANG

3 Contoh Masukan/Keluaran

Masukan	Keluaran
10	
TAMBAH 101 Beras 50 15000	BARANG DITAMBAHKAN
TAMBAH 205 Gula 25 12000	BARANG DITAMBAHKAN
TAMBAH 101 Tepung 30 8000	KODE SUDAH ADA
CARI 101	101 Beras 50 15000
CARI 999	BARANG TIDAK DITEMUKAN
JUAL 101 10	BERHASIL TERJUAL
JUAL 101 50	STOK TIDAK CUKUP
RESTOK 205 20	STOK DITAMBAHKAN
RESTOK 999 10	BARANG TIDAK DITEMUKAN
LAPORAN	101 Beras 40 15000 205 Gula 45 12000

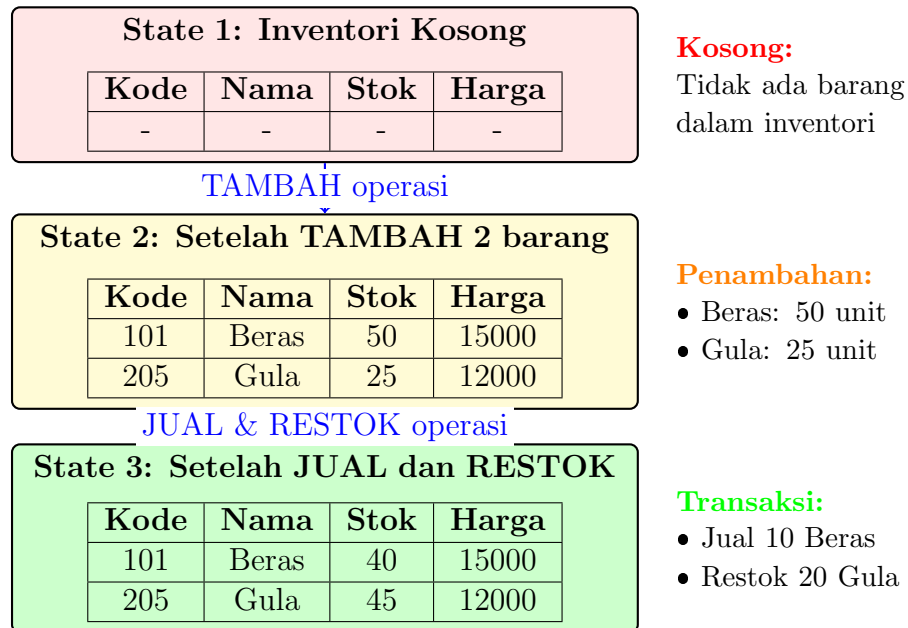
4 Penjelasan

Mari kita ikuti alur operasi pada contoh di atas:

1. **TAMBAH 101 Beras 50 15000**: Menambahkan barang baru dengan kode 101 (Beras, stok 50, harga 15000). Berhasil ditambahkan.
2. **TAMBAH 205 Gula 25 12000**: Menambahkan barang baru dengan kode 205 (Gula, stok 25, harga 12000). Berhasil ditambahkan.
3. **TAMBAH 101 Tepung 30 8000**: Mencoba menambahkan barang dengan kode 101, tapi kode sudah digunakan oleh Beras. Gagal.
4. **CARI 101**: Mencari barang dengan kode 101. Ditemukan: Beras dengan stok 50 dan harga 15000.
5. **CARI 999**: Mencari barang dengan kode 999. Tidak ditemukan.
6. **JUAL 101 10**: Menjual 10 unit Beras. Stok mencukupi (50 - 10), sehingga stok berkurang menjadi 40.
7. **JUAL 101 50**: Mencoba menjual 50 unit Beras. Stok tidak cukup (40 < 50), sehingga gagal.
8. **RESTOK 205 20**: Menambah stok Gula sebanyak 20 unit. Stok bertambah dari 25 menjadi 45.
9. **RESTOK 999 10**: Mencoba menambah stok barang kode 999. Barang tidak ditemukan.
10. **LAPORAN**: Menampilkan semua barang diurutkan berdasarkan kode:
 - Kode 101: Beras, stok 40, harga 15000
 - Kode 205: Gula, stok 45, harga 12000

5 Ilustrasi Sistem

Sistem Inventori Toko Pak Budi



6 Catatan Implementasi

- Gunakan array atau list untuk menyimpan data barang
- Implementasikan pencarian linear untuk operasi CARI
- Gunakan algoritma sorting sederhana (bubble sort/selection sort) untuk LAPORAN
- Perhatikan penanganan kasus edge: barang tidak ditemukan, stok habis, dll.
- Pastikan validasi input untuk mencegah error
- Kompleksitas waktu yang diharapkan: $O(n)$ untuk pencarian, $O(n^2)$ untuk sorting