Jobsheet 7

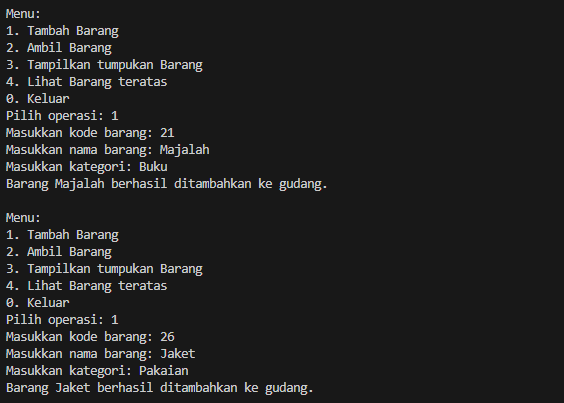
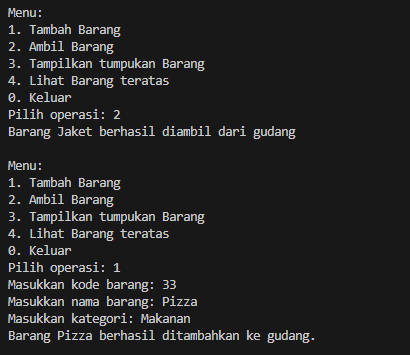
Praktikum Algoritma & Struktur Data

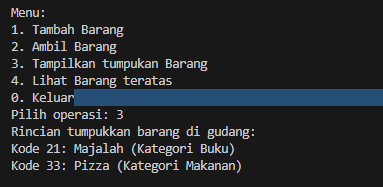


Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 2  
2024

**SS Hasil Percobaan 1**

****

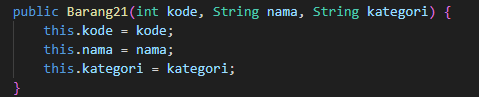


**Pertanyaan Percobaan 1**

1. **Lakukan perbaikan pada kode program, sehingga keluaran yang dihasilkan sama dengan verifikasi hasil percobaan! Bagian mana saja yang perlu diperbaiki?**

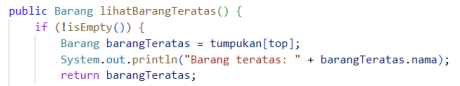
**Jawab:**

* Menambahkan konstruktor berparameter pada class Barang21, agar dapat menginputkan data barang.

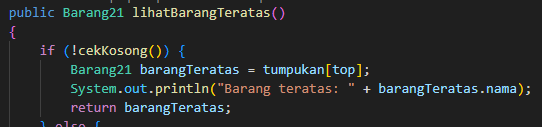


* Menubah !isEmpty menjadi !cekKosong pada method lihatBarangTeratas dikarenakan tidak terdapat method isEmpty.

Before:



After:

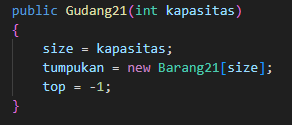


1. **Berapa banyak data barang yang dapat ditampung di dalam tumpukan? Tunjukkan potongan kode programnya!**

**Jawab:**

Pada class Gudang21 terdapat konstruktor yang memiliki parameter kapasitas, kemudian konstruktor tersebut dipanggil pada main dan parameter kapasitas tersebut wajib diisi. Di dalam class main sendiri kapasitas memiliki nilai 7, sehingga banyak data yang ditampung adalah 7.

Konstruktor berparameter kapasitas:



Kode yang berisi nilai banyaknya data:



1. **Mengapa perlu pengecekan kondisi !cekKosong() pada method tampilkanBarang? Kalau kondisi tersebut dihapus, apa dampaknya?**

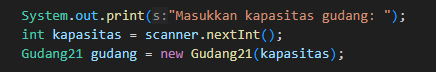
**Jawab:**

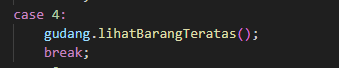
Pengecekan kondisi !cekKosong pada method tampilkanBarang() diperlukan untuk memastikan bahwa tumpukan barang tidak kosong sebelum mencoba untuk menampilkan barang-barangnya. Dampak ketika kita hapus kondisi tersebut adalah program akan mencoba melakukan iterasi melalui tumpukkan array dan mencoba untuk mengakses atribut dari Barang21.

1. **Modifikasi kode program pada class Utama sehingga pengguna juga dapat memilih operasi lihat barang teratas, serta dapat secara bebas menentukan kapasitas gudang!**

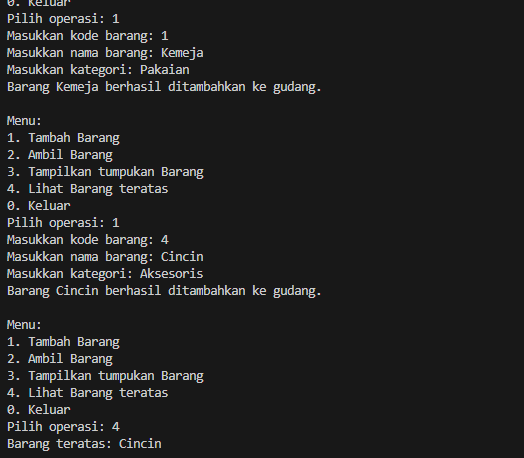
**Jawab:**

Code:





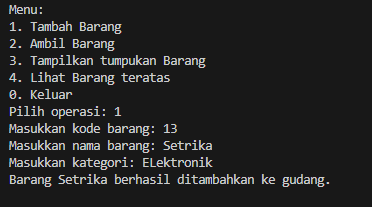
Output:

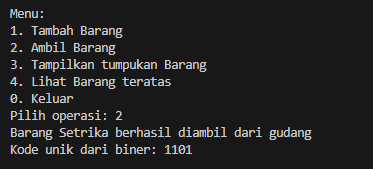


1. **Commit dan push kode program ke Github.**

<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%207/Percobaan%201>

**SS Hasil Percobaan 2**





**Pertanyaan Percobaan 2**

1. **Pada method konversiDesimalKeBiner, ubah kondisi perulangan menjadi while (kode != 0), bagaimana hasilnya? Jelaskan alasannya!**

**Jawab:**

Hasilnya akan tetap sama. Alasan utama mengapa menggunakan kondisi while (kode != 0) lebih fleksibel adalah karena itu memungkinkan penanganan nilai kode yang negatif.

1. **Jelaskan alur kerja dari method konversiDesimalKeBiner!**

**Jawab:**

Program akan melakukan operasi modulo 2 dengan variabel int kode, kemudian hasil sisa modulo 1/0 akan disimpan di stack sedangkan hasil modulo akan disimpan di variabel kode. Dan program akan terus melakukan iterasi hingga variabel kode bernilai 0.

1. **Push & commit github.**

<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%207/Percobaan%202>

**SS Hasil Percobaan 3**



**Pertanyaan Percobaan 3**

1. **Pada method derajat, mengapa return value beberapa case bernilai sama? Apabila return value diubah dengan nilai berbeda-beda setiap case-nya, apa yang terjadi?**

**Jawab:**

Return value bernilai sama yang menentukan derajatnya berarti sama, jika setiap nilai return berbeda, susah untuk mengelompokkannya untuk membandingkan inputan char dan operasi.

1. **Jelaskan alur kerja method konversi!**

**Jawab:**

* Jika ‘c’ operand maka, menambahkan c ke P.
* Jika ‘(‘, panggil fungsi push( c ) untuk memasukannya ke stack.
* Jika ‘)’, melakukan iterasi selama karakter teratas stack bukan ‘(‘, mengeluarkan karakter teratas dari stack menggunakan pop() dan menambahkannya ke P.
* Jika ‘c’ operator maka, melakukan iterasi selama prioritas lebih besar sama dengan antara derajat c dengan derajat stack teratas dan mengeluarkan karakter teratas dari stack menggunakan pop() dan tambahkannya ke P. Setelah itu memasukkan karakter c ke stack.
* Akhirnya mengembalikan nilai P.

1. **Pada method konversi, apa fungsi dari potongan kode berikut?**



**Jawab:**

Potongan kode tersebut digunakan untuk mengambil karakter pada posisi indeks ke-i dari string Q dan menyimpannya ke dalam variabel c.

1. **Push & Commit Github**

<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%207/Percobaan%203>

**Latihan Praktikum**

**Perhatikan dan gunakan kembali kode program pada Percobaan 1. Tambahkan dua method berikut pada class Gudang:**

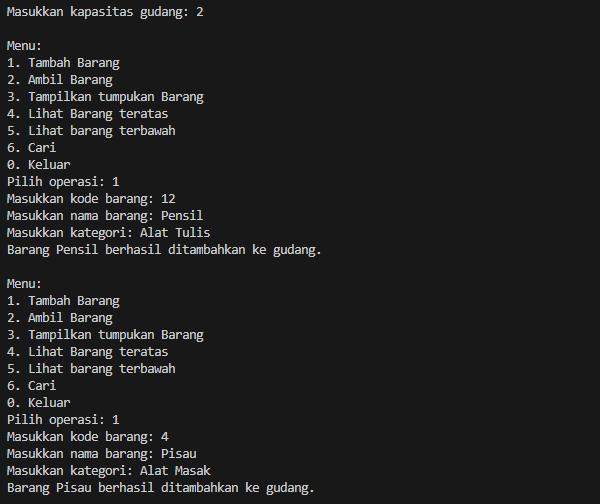
* **Method lihatBarangTerbawah digunakan untuk mengecek barang pada tumpukan terbawah.**
* **Method cariBarang digunakan untuk mencari ada atau tidaknya barang berdasarkan kode barangnya atau nama barangnya.**

Source Code:

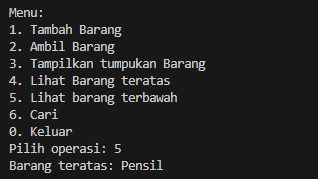
<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%207/Latihan%20Praktikum>

Output:

Input data barang:



Lihat barang terbawah:



Mencari barang berdasarkan kode dan nama barang:

