

SISTEM ANTRIAN *SINGLE CHANNEL SINGLE SERVER* DENGAN METODE *POISSON* MENGGUNAKAN STRUKTUR *CIRCULAR ARRAY* STUDI KASUS: ANTRIAN NASABAH BANK

Ade Supriatna

Teknik Informatika, STMIK Subang
Jalan Marsinu No. 5 Tegal Kalapa Subang
email: adespr@yahoo.com

Abstrak – Istilah antrian (*queue*) senantiasa kita jumpai di tempat-tempat yang dijadikan layanan publik, baik sektor pemerintah maupun swasta. Penelitian ini mengambil objek perbankan sebagai suatu lembaga yang menggunakan sistem antrian dalam pelayanan kepada nasabahnya. Sistem antrian bila tidak ditata dengan teratur, ditambah tidak ada disiplin pengantri (yang populer disebut *First Come, First Serve*) akan mengakibatkan kesemrawutan antrian bahkan lebih fatalnya akan menyebabkan kehilangan pelanggan. Seiring perkembangan teknologi informasi maka sistem antrian konvensional diubah menjadi sistem antrian berbasis komputer. Dalam penelitian ini tata letak fisik sistem antrian adalah *single channel single server*. Metode yang digunakan adalah distribusi *Poisson*. Metoda distribusi probabilitas *Poisson* adalah metoda dimana kedatangan diasumsikan terjadi dengan kecepatan rata-rata yang konstan dan bebas satu sama lain. Membahas tentang sistem antrian dengan bantuan komputer artinya melibatkan struktur tempat penyimpanan data. Struktur data menggunakan larik memutar (*circular array*). Antarmuka aplikasi dibuat dengan salah satu bahasa pemrograman tingkat tinggi yakni bahasa pemrograman *Delphi*. Harapan dari penelitian ini adalah adanya efektifitas dan efisiensi dari sistem antrian di lembaga perbankan sehingga berdampak pada kenyamanan layanan nasabah.

Kata Kunci: *Antrian, Single channel Single server, Poisson, Circular Array*

I. PENDAHULUAN

Apa sebenarnya latar belakang dari perlunya sistem antrian secara umum. Fakta di lapangan bahwa mekanisme antrian konvensional cenderung mengakibatkan, peserta antrian saling serobot untuk mendapatkan layanan yang paling awal, hal ini akan membuat suasana tidak tertib. Kadang-kadang sering terjadi keributan gara-gara orang terakhir datang, ingin dilayani paling awal, dengan alasan prioritas, diburu waktu, dan sebagainya.

Pada umumnya sistem antrian misalnya di bank sudah menerapkan perangkat teknologi informasi. Akan tetapi di beberapa cabang masih terlihat adanya pelayanan nasabah yang masih mengabaikan disiplin antrian *first come first serve*, dengan pertimbangan kedatangan antrian nasabah tidak terlalu banyak dan rentang waktu kedatangan nasabah cukup longgar.

Struktur data menyediakan suatu tempat penyimpanan data yang disusun sedemikian rupa sehingga memudahkan dalam mengakses data. Salah satu struktur data yang populer adalah larik (*array*). Penambahan dan penghapusan data yang biasa terjadi dalam struktur larik apabila tidak disusun dengan baik akan mengakibatkan adanya tempat yang kosong, sementara sesungguhnya masih bisa digunakan untuk menyimpan data antrian.

Sudah waktunya kita memanfaatkan teknologi komputer untuk mengatur mekanisme antrian. Oleh karena itu penelitian ini mengambil topik “Sistem Antrian *Single Channel Single Server* Dengan Metode *Poisson* Menggunakan Struktur *Circular Array* Studi kasus : Antrian Nasabah Bank”.

II. LANDASAN TEORI

“Sistem antrian memiliki tiga komponen yaitu kedatangan populasi, antrian, fasilitas pelayanan”[3]. Kedatangan populasi nasabah bank termasuk populasi tidak terbatas (*infinite*) hal ini akan menimbulkan panjang antrian.

“Disiplin antrian (*queue*) yakni pertama datang, pertama dilayani (*First Come First Serve*) atau istilah lainnya adalah *FIFO* atau *First In First Out*”[4], dapat diartikan bahwa data pertama masuk kedalam struktur penyimpanan data misalnya larik, akan diperlakukan yang pertama pula yang akan diproses. Metoda akses *FIFO* dalam larik akan mengakibatkan pergeseran penempatan data pada indeks larik supaya ruang pertama dalam larik senantiasa terisi oleh data berikutnya yang berada dalam posisi antrian. Masalah yang terjadi adalah proses pergeseran membutuhkan langkah yang panjang. Masalah lain adalah larik adalah struktur data statis, dimana elemen larik tidak dapat ditambah pada saat proses berlangsung, sementara data/informasi

antrian terus bertambah.

Banyaknya jalur antrian dapat disesuaikan dengan volume antrian, misalnya ketika jumlah pengantri masih dikategorikan relatif sedikit maka cukup digunakan satu lajur satu layanan (*Single Channel Single Server*)[4].

Distribusi probabilitas *Poisson* bahwa suatu kedatangan dalam antrian digambarkan dengan variabel acak yang terputus-putus dan bukan bilangan bulat negatif [3]. Diambil sampel dari sebuah bank bahwa dalam antrian selama 30 menit nasabah yang antri bisa 6, 8, dst. Probabilitas n kedatangan dalam waktu T ditentukan dengan rumus :

$$P(n,T) = \frac{e^{-T} (T)^n}{n!} \text{ dimana } n=0,1,2,\dots \text{dst. [1]}$$

Keterangan :

λ = rata-rata kedatangan persatuan waktu

T = periode waktu

n = jumlah kedatangan dalam waktu T

$P(n,T)$ = probabilitas n kedatangan dalam waktu T

Probabilitas prestasi dapat dilihat dari : rata-rata jumlah kedatangan, rata-rata waktu tunggu dari suatu kedatangan dan persentase waktu luang dari pelayanan.

Ukuran prestasi kemudian dapat digunakan untuk memutuskan jumlah pelayanan yang harus diberikan, perubahan yang harus dilakukan dalam kecepatan pelayanan atau perubahan lain dalam sistem antrian. Dengan sasaran pelayanan, jumlah pelayan dapat ditentukan tanpa berpatokan pada biaya waktu tunggu. Ukuran prestasi dan parameter model antrian ditentukan dengan notasi sebagai berikut:

λ = rata-rata kecepatan/jml kedatangan (persatuan waktu)

$1/\lambda$ = rata-rata waktu antar kedatangan

μ = rata-rata kecepatan pelayanan (jumlah satuan yang dilayani persatuan waktu bila pelayan sibuk).

$1/\mu$ = rata-rata waktu yang dibutuhkan pelayan

p = faktor penggunaan pelayan (proporsi waktu pelayan ketika sedang sibuk)

P_n = probabilitas bahwa n satuan (kedatangan) dalam sistem

L_q = rata-rata jumlah satuan dalam antrian (rata-rata panjang antrian)

L_s = rata-rata jumlah satuan dalam sistem

W_q = rata-rata waktu tunggu dalam antrian

W_s = rata-rata waktu tunggu dalam sistem

Dalam penelitian ini akan dibahas permasalahan antrian yang didasarkan pada asumsi berikut :

1. Satu pelayanan dan satu tahap.
2. Jumlah kedatangan per unit waktu digambarkan oleh Distribusi Poisson dengan λ = rata-rata kecepatan kedatangan
3. Waktu pelayanan eksponensial dengan μ = rata-rata kecepatan pelayanan
4. Disiplin antrian adalah first come first served (Aturan antrian pertama datang-pertama

dilayani) seluruh kedatangan dalam barisan hingga dilayani,

5. dimungkinkan panjang barisan yang tak terhingga.

6. populasi yang dilayani tidak terbatas

7. rata-rata kedatangan lebih kecil dari rata-rata waktu pelayanan

Dari asumsi tersebut dapat diperoleh hasil secara statistik sebagai berikut :

P_w = probabilitas fasilitas layanan sibuk atau faktor utilisasi fasilitas

$$= \lambda / \mu$$

L_q = jumlah rata-rata dalam antrian

$$L_q = \frac{\lambda^2}{\mu(\mu-\lambda)}$$

L_s = jumlah rata-rata di dalam sistem (yang antri dan yang sedang dilayani)

$$L_s = \frac{\lambda}{\mu - \lambda}$$

W_q = waktu rata-rata di dalam antrian

$$W_q = \frac{\lambda}{\mu(\mu-\lambda)}$$

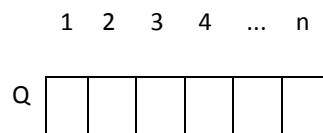
W_s = waktu rata-rata di dalam sistem

$$W_s = \frac{1}{\mu - \lambda}$$

Definisi struktur data adalah *A data structure is a set of domain, a designated domain, a set of functions, and a set of axioms* [2]. Struktur data sangat erat kaitannya dengan algoritma. Definisi algoritma adalah “urutan logis pengambilan putusan untuk pemecahan masalah” [5].

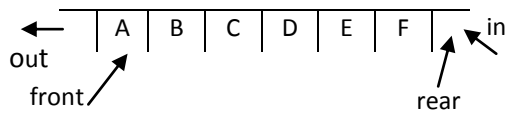
Salah satu materi bahasan dalam algoritma dan struktur data adalah tentang antrian. Definisi antrian adalah “suatu kumpulan data yang mana penambahan elemen hanya bisa dilakukan pada suatu ujung disebut dengan sisi belakang atau *rear*, dan penghapusan/pengambilan elemen dilakukan lewat ujung lain disebut sisi depan atau *front*” [5].

Larik adalah tipe terstruktur dimana setiap komponen diacu melalui indeks [7]. Larik dapat diilustrasikan sebagai deretan/tumpukan kotak :



Gambar 2.1 Larik bernama Q

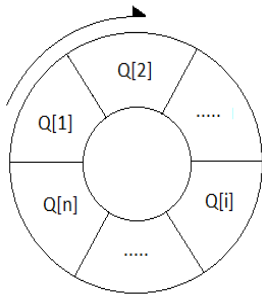
Kemudian Larik dapat dipergunakan untuk antrian, yakni terdapat bagian depan (*front*), bagian belakang (*rear*), elemen pertama masuk akan diperlakukan sebagai elemen yang pertama keluar, berikut ini larik dengan asumsi tipe elemen adalah karakter.



Gambar 2.2 Ilustrasi Antrian dengan 6 elemen

Larik memutar (*circular array*) berasal dari larik biasa yang bersifat statis, akan tetapi pengaksesan elemen larik antara awal larik dan ujung larik diposisikan seolah-olah menyambung. Larik pada dasarnya merupakan sebuah struktur data.

Bentuk larik secara standar adalah lurus (*straight*). Akan tetapi dapat dibuat sebagai larik memutar (*circular*), seperti gambar berikut ini :



Gambar 2.3 Ilustrasi Larik Memutar (*Circular Array*)[7].

Diagram konteks dan data flow diagram merupakan suatu teknik penggambaran untuk pemodelan analisa sistem secara logika[6]. Dalam diagram konteks ada simbol lingkaran sebagai representasi dari sistem/ perangkat lunak yang dibangun, simbol kotak/ persegi sebagai representasi dari entitas luar yang terlibat terhadap sistem, dan garis sebagai representasi dari aliran data/informasi. Dekomposisi proses-proses yang terjadi dalam sistem digambarkan dalam *data flow diagram* dan spesifikasi proses.

III. PEMBAHASAN

3.1 Analisa

Dengan rumusan teori distribusi Poisson yang dibahas di landasan teori secara ringkas dapat diuraikan berikut ini dengan contoh asumsi dalam 30 menit jumlah nasabah yang datang 10 orang, seorang petugas layanan bisa melayani sampai 12 orang per 30 menit. Maka didapat rumusan perhitungan ukuran prestasi sbb :

No.	Statistik ukuran prestasi	Nilai
1	Rata-rata jumlah nasabah dalam antrian (L_q)	4,17
2	Rata-rata jumlah nasabah dalam sistem (L_s)	5
3	Rata-rata waktu nasabah dalam antrian (W_q)	0,42
4	Rata-rata waktu nasabah dalam	0,5

	sistem (W_s)	
5	Probabilitas fasilitas layanan sibuk (P_w)	0,83

Data-data hasil distribusi Poisson tersebut dapat dijadikan acuan apakah efektif atau tidak dari layanan terhadap jumlah antrian yang terjadi.

Konsep antrian pada *circular array* adalah penempatan data atau informasi yang melibatkan penambahan ke antrian serta pengeluaran (penghapusan) dari antrian. Penghapusan tersebut mengakibatkan adanya ruang kosong. Pada *array* biasa (lurus) awal antrian dan akhir antrian seolah-olah terpisah bahkan berjauhan. Pada *array* yang memiliki 10 elemen misalnya, maka apabila terjadi pemrosesan baru 2 kali, maka seolah-olah data pada 2 elemen pertama dianggap kosong. Apabila terjadi penambahan ke antrian tidak bisa karena elemen maksimum *array* hanya sepuluh, padahal ada 2 ruang kosong bekas 2 antrian pertama. Bisa saja kita melakukan penggeseran data, akan tetapi kalau dilakukan berarti merubah ulang semua posisi elemen *array*, hal ini tidak efisien. Solusinya adalah dengan memanfaatkan ruang kosong dari antrian yang telah diproses. Jadi seolah-olah kita memaksa melingkarkan bentuk larik menjadi seperti berputar (lingkaran). Apabila bentuk *array* ini menjadi memutar, maka akan nampak seolah-olah elemen pertama *array* bertautan dengan elemen terakhir *array*. Dengan adanya ruang kosong tersebut maka akan digunakan kembali ketika untuk penempatan data/informasi berikutnya.

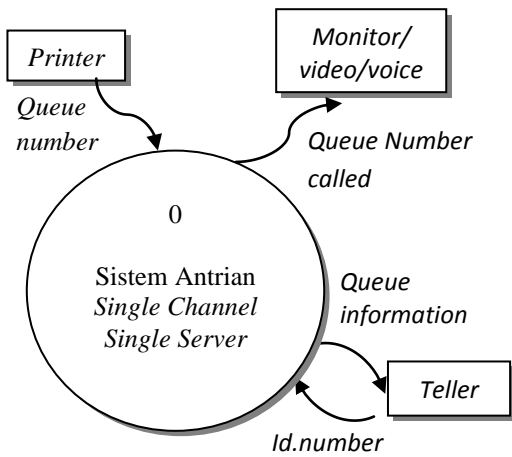
Deklarasi struktur antrian dengan larik memutar dapat dilihat berikut ini:

```
Const max_elemen=tentukan_nilai_maksimum
Type antri = array[1..max_elemen] of integer
Var antrian : antri
    depan,belakang : integer
```

3.2 Pemodelan

Context Diagram

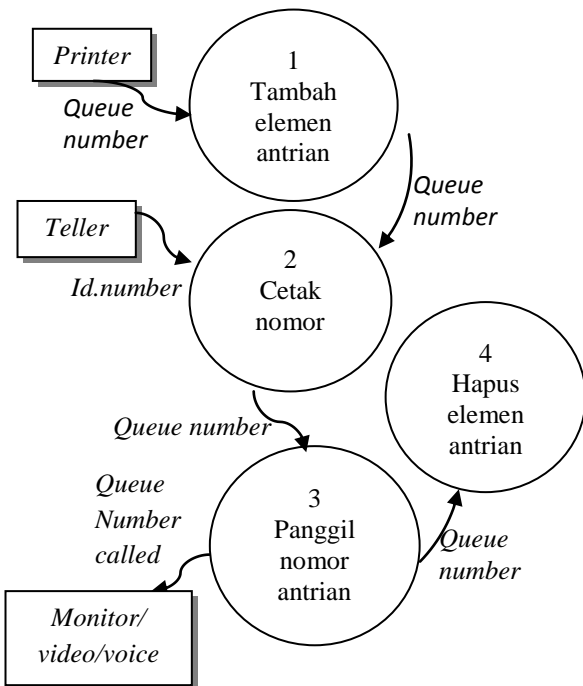
Sistem bernama: Sistem Antrian *Single Channel Single Server*. Terdiri dari entitas luar Printer, Monitor/ video/ voice, Teller. Seorang nasabah ketika memasuki antrian dimulai dengan memilih nomor layanan dalam interface/ panel yang akan terhubung dengan printer untuk mencetak nomor antrian. Nasabah menunggu panggilan layanan. Nasabah mendapat giliran layanan yang dilayani oleh Teller.



Gambar 3.1 : Diagram Konteks Sistem Antrian dengan Circular Array

Data Flow Diagram

Data flow Diagram merupakan dekomposisi dari context diagram. Karena context diagram merupakan gambaran sistem bersifat generik, maka perlu penjabaran secara detil setiap proses yang terjadi dalam sistem dan digambarkan dalam DFD. Proses-proses yang terjadi dalam sistem meliputi tambah elemen antrian, cetak nomor, panggil nomor antrian, hapus elemen antrian.



Gambar 3.2 : Data Flow Diagram Sistem Antrian dengan Circular Array

Spesifikasi Proses

Nomor Proses : 1
 Nama Proses : Tambah elemen antrian
 Algoritma:

```

If belakang:=max_elemen then belakang:=1
Else belakang:=belakang+1
endif
If depan=belakang then
Write('antrian sudah penuh')
Else Q[belakang]:=X
endif
    
```

Nomor Proses : 2
 Nama Proses : Cetak nomor
 Algoritma :
 Aktifkan printer/ printer is on
 If kosong(Q) then
 Write('Antrian kosong...')
 Else
 Cetak(Q)
 Endif

Nomor Proses : 3
 Nama Proses : Panggil nomor antrian
 Algoritma :
 Aktifkan mode suara
 Write(Q[depan])

Nomor Proses : 4
 Nama Proses : Hapus elemen antrian
 Algoritma :
 If kosong(Q) then
 Write('Antrian kosong...')
 Else
 Hapus:=Q[depan]
 If depan:= max_elemen then
 Depan:=1
 Else
 Depan:=depan+1
 Endif
 Endif

3.3 Perancangan

Perancangan meliputi antarmuka. Perancangan antarmuka sistem antrian yang menggambarkan posisi nomor antrian, jenis layanan, waktu sistem. Selain display juga dilengkapi dengan audio/ voice yang terhubung ke speaker.

Hasil rancangan antarmuka ditindaklanjuti dengan perancangan antarmuka user dengan bahasa pemrograman.

Paradigma pengembangan perangkat lunak dapat menggunakan salah satu model yang direferensikan oleh Pressman[6].

3.4 Implementasi

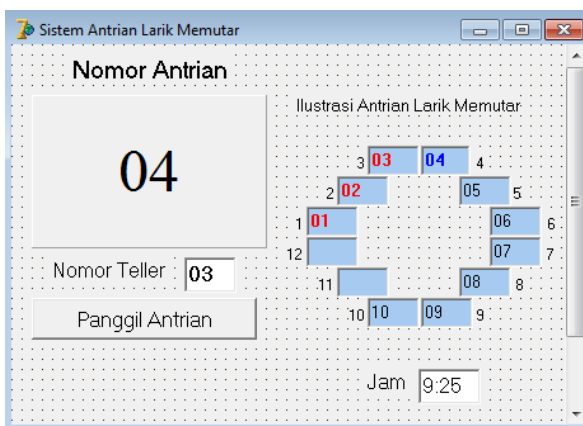
Bahasa pemrograman untuk mengimplementasikan struktur larik antrian adalah Borland Delphi7. Bahasa pemrograman ini merupakan bahasa pemrograman general purpose. Delphi dikembangkan oleh Borland Inc. dan termasuk kedalam teknik pemrograman berbasis komponen, serta diterapkan untuk interface grafik.

Berikut adalah sebagian kode program untuk menambah ke circular array :

```

procedure tambah(var antrian:antri;no_antri:integer);
begin
  if belakang=max_elm then belakang:=0;
  else belakang:=belakang+1;
  if not(kosong(antrian)) then
  begin
    antrian(belakang):=no_antri;
  end
  else
  begin
    messagedlg('sudah penuh');
    belakang:=belakang-1;
    if belakang=0 then
      belakang:=max_elm;
    end;end;end;end;

```



Gambar 3.3 : Tampilan Interface sistem antrian

IV. KESIMPULAN

Array merupakan struktur data statis, akan tetapi dapat dimanfaatkan dengan optimal untuk menampung data yang bersifat “dinamis terukur” (artinya jumlah data bisa jadi berkembang akan tetapi dalam jumlah terbatas), dengan kata lain adanya asumsi awal bahwa rata-rata setiap hari terdapat data antrian sebanyak n antrian, walaupun bertambah atau

berkurang tidak terlalu bermasalah dengan tempat (*array*) yang sudah disiapkan.

Array dalam penelitian ini diperlakukan seperti sebuah untaian yang melingkar menghubungkan antara bagian awal *array* dan bagian akhir *array*. Melalui pola *array* melingkar tersebut akan lebih memudahkan dalam penempatan data pada lokasi-lokasi yang masih kosong dari sebuah struktur data *array* sebagai akibat dari adanya proses penghapusan dari suatu antrian.

Metode probabilitas *Poisson* akan membantu menghitung efektifitas pemakaian waktu dalam antrian, sehingga dapat dijadikan acuan apakah layanan perlu ditambah atau tidak ketika kedatangan antrian meningkat.

DAFTAR REFERENSI

- [1] Basuki, Ahmad dkk., “Modeling dan Simulasi”, Penerbit: IPTAQ Mulia Media, Jakarta, 2004..
- [2] Horowitz, Ellis, Sartaj Sahni, “Fundamentals of Data Structures”, Computer Science Press Inc., Rockville, Maryland, 1981
- [3] <http://sutanto.staff.uns.ac.id>
- [4] Kakiay, Thomas J., “Dasar Teori Antrian untuk Kehidupan Nyata”, Penerbit: Andi, Yogyakarta, 2004.
- [5] Munir, Rinaldi, “Buku Teks Ilmu Komputer Algoritma & Pemrograman Dalam Bahasa Pascal dan C”, Penerbit: Informatika, Bandung, 2003.
- [6] Pressman, Roger S., “Software Engineering : A Practitioner’s Approach Sixth Edition”, Published by Mc Graw Hill, New York, 2005.
- [7] Santosa, P. Insap, “Struktur Data Menggunakan Turbo Pascal 6.0”, Andi Offset, Yogyakarta, 1993.

Biodata Penulis

Ade Supriatna, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika STMIK Mardira Indonesia, di Bandung, lulus tahun 2001. Memperoleh gelar Magister Teknik (M.T.) Program Pasca Sarjana Magister Informatika STEI - ITB, lulus tahun 2006. Saat ini menjadi Dosen Kopertis IV dpk. pada STMIK Subang.