

IMPLEMENTASI ALGORITMA B-TREE UNTUK PENCARIAN KELAS PENGGANTI PADA UNIVERSITAS BUNDA MULIA

Lukman Hakim, Reagen
lhakim@bundamulia.ac.id, Reagen_ragn@yahoo.co.id
Teknik Informatika Universitas Bunda Mulia

Abstract

Pengendalian proses pembelajaran atau perkuliahan pada setiap perguruan tinggi sangat penting, selain meningkatkan mutu pembelajaran serta ketercapaian kehadiran minimal 80% dari setiap matakuliah, kehadiran perkuliahan merupakan syarat yang harus tercapai dalam borang akreditasi. Aplikasi penjadwalan kelas pengganti ini membantu layanan akademik untuk mengolah kehadiran dosen dalam perkuliahan dan penentuan jadwal perkuliahan pengganti. Pengolahan dan pengendalian perkuliahan serta penentuan kelas pengganti (*Makeup Class*) menjadi kendali dalam penentuan kelas dan mahasiswa yang bentrok pada saat kelas pengganti dijadwalkan, Sistem Informasi Akademik mengimplementasi Algoritma B-Tree pada sistem penentuan jadwal perkuliahan pengganti dapat mengurangi terjadinya kesalahan penjadwalan atau bentrok terhadap kelas pengganti yang telah diambil sebelumnya.

Kata Kunci: *B-tree, Aplikasi Penjadwalan Kelas*

PENDAHULUAN

Pentingnya kehadiran tatap muka ataupun *online* pada setiap perkuliahan berdasarkan pemerintah atau DIKTI mengharapkan minimal 80% kehadiran setiap matakuliah, hal ini untuk mendorong untuk ketercapaian penyampaian materi kuliah secara baik serta menghindari terjadinya ketidak tercapaian proses pembelajaran. Proses pengendalian perkuliahan dosen sangat penting untuk menjamin mutu pembelajaran setiap perguruan tinggi atau Universitas dengan terkendalinya proses kehadiran dosen serta mekanisme kuliah pengganti secara lebih baik. Aplikasi kehadiran dan penjadwalan kuliah pengganti memberikan kemudahan untuk pengendalian dan penjadwalan perkuliahan tidak sesuai dengan jadwal secara akurat dalam hal penentuan kelas atau lab serta mengurangi bentrok mahasiswa yang mengikuti matakuliah tersebut dengan matakuliah yang sudah terjadwal secara lebih efektif. Penerapan algoritma B-Tree pada proses pencarian kelas, Matakuliah, Mahasiswa, waktu pada

aplikasi penentuan kuliah pengganti memberikan mudahan dan akurat.

Tujuan penelitian ini merancang bangun aplikasi penentuan kuliah pengganti dengan algoritma B-Tree.

Manfaat penelitian ini membantu layanan akademik/perkuliahan dalam pengendalian perkuliahan terjadwal atau tidak terjadwal atau kuliah pengganti lebih efektif.

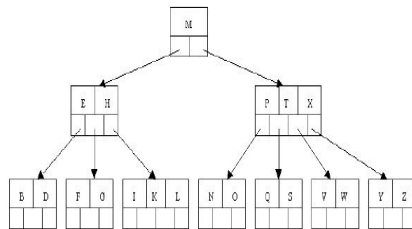
Algoritma B-Tree

B-Tree adalah sebuah *m-ary balanced search tree* khusus yang digunakan dalam basis data karena strukturnya memungkinkan data yang disimpan untuk disisipi, dihapus, dan diambil dengan jaminan proses dengan waktu terburuk, di mana setiap simpulnya terdiri dari $(m/2)$ sampai m buah simpul anak, di mana $m > 1$ merupakan bilangan bulat. m adalah orde. Akar pohon B-tree paling sedikit memiliki 2 simpul anak. Ini adalah struktur yang baik jika pohon digunakan pada memori yang lambat, karena ketinggian dan jumlah akses dapat

diperkecil dengan mengambil bilangan m yang besar.

Balanced tree atau pohon seimbang adalah pohon dimana tidak ada simpul daun yang lebih panjang terhadap daun yang lain. *Search tree* adalah pohon dimana setiap subpohon dari sebuah simpul mempunyai kunci lebih kecil dari subpohon kanan simpul tersebut. Kunci dalam sebuah simpul secara konsep berada di antara subpohon-subpohon dan lebih besar dari kunci di subpohon kiri simpul dan lebih kecil dari kunci di subpohon kanan simpul.

Sebuah B-Tree didesain untuk digunakan pada disk. Disk hanya dapat membaca dan menulis blok data ukuran tetap (berukuran besar) sekaligus. Sebuah B-tree menyimpan banyak kunci di setiap simpulnya sehingga (1) sebuah disk dapat mengakses banyak kunci, dan (2) faktor cabang pohon sangat tinggi (dalam prakteknya lebih besar dari 1000) sehingga pohon dengan ketinggian kecil dapat menyimpan kunci dalam jumlah yang sangat besar, yang dapat diakses hanya dengan beberapa operasi. Untuk lebih jelasnya ditunjukkan pada gambar 1.



Gambar 1. Pohon B-tree

Normalisasi

Menurut Al Bahra bin (2013, p168), Proses normalisasi pertama kali diperkenalkan oleh E.F.Codd pada tahun 1972. Normalisasi sering dilakukan sebagai suatu uji coba pada suatu relasi secara berkelanjutan untuk menentukan apakah relasi tersebut sudah baik atau masih melanggar aturan - aturan standar yang dibelakukan pada suatu relasi yang normal (sudah dapat dilakukan proses

insert, update, delete, dan modify pada satu atau beberapa atribut tanpa mempengaruhi integritas data dalam relasi tersebut).

Proses normalisasi merupakan metode yang formal/standard dalam mengidentifikasi dasar relasi bagi primary keynya (atau *candidat key* dalam kasus BCNF), dan dependensi fungsional diantara atribut - atribut dari relasi tersebut. Normalisasi akan membant perancang basis data dengan menyediakan suatu ujicoba yang berurut yang dapat diimplementasikan pada hubungan individual, sehingga skema relasi dapat dinormalisasi ke dalam bentuk yang lebih spesifik untuk menghindari terjadinya error atau inkonsistensi data, bila dilakukan update terhadap relasi tersebut dengan Anomaly.

Macam - macam Anomaly / penyimpangan:

Insertion Anomaly

Menurut Al Bahra bin (2013, p171), Insertion Anomaly merupakan error atau kesalahan yang terjadi sebagai akibat dari operasi meyisipkan (*insert*) *tuple/record* pada sebuah relasi

Delete Anomaly

Menurut Al Bahra bin (2013, p171), Delete Anomaly merupakan error atau kesalahan yang terjadi sebagai akibat dari operasi penghapusan (*delete*) *tuple/record* pada sebuah relasi

Update Anomaly

Menurut Al Bahra bin (2013, p171), Insertion Anomaly merupakan error atau kesalahan yang terjadi sebagai akibat dari operasi perubahan (*update*) *tuple/record* pada sebuah relasi

Pembahasan - Normalisasi data

Data tak Normal

Data tak normal ini didapat dari proses kegiatan perkuliahan dan pemantauan daftar hadir dosen pada

Universitas XYZ. Untuk lebih jelasnya ditunjukkan pada tabel 1

Tabel 1. Tabel Unnormal

Kelas	NamaDosen	Jam	Ruang	Tanggal	MataKuliah
3PT13	Christin	11.30-13.10	R302	20-01-2014	Basis Data
3PT13	Christin	15.30-17.10	Lab-A	20-01-2014	Basis Data
3PS11	Boy Firmansyah	11.30-13.10	R602	21-01-2014	Komunikasi Data
3PS11	Boy Firmansyah		R602	21-01-2014	Komunikasi Data
3PT11	Asri Partiw		R503	19-01-2014	Kecerdasan Buatan
3PT11	Asri Partiw		R503	19-01-2014	Kecerdasan Buatan

Tabel Normal ke satu

Tabel 2. 1NF dibawah didapat dari tabel tidak normal yang telah dinormalkan dan diberikan primary key untuk sebagai penanda setiap data. Tabel 1NF dibawah akan dijadikan bentuk normal kedua agar data yang ada dapat lebih jelas, saat pencarian data yang ada lebih mudah. Untuk lebih jelasnya dapat dilihat pada tabel 2.

Tabel 2. Normal ke satu

Id	Tanggal	NamaDosen	Kelas	Mata kuliah	Ruang	Jam
1	28/03/2014	Christin	3PT11	Perancangan Web	R301	07.30-11.10
2	28/03/2014	Christin	1PT11	Algoritma	R410	07.30-11.10
3	27/03/2014	Halim Agung	1PMN3	Pengantar T.Info	R601	09.30-11.10
4	27/03/2014	Halim Agung	1PMN2	Pengantar T.Info	R602	11.30-13.10

Tabel Normal kedua (2NF)

Berdasarkan tabel normal kesatu salah satu tabel bentuk normal kedua adalah tabel kelas yang berisi kelas aktif yang ada dalam perkuliahan. Untuk lebih jelasnya terdapat pada tabel 3.

Tabel 3. Kelas

ID Kelas	Kelas
1PT11	1PT11
1PT12	1PT12

Berdasarkan tabel normal kesatu dipecah menjadi salah satu tabel bentuk normal kedua adalah tabel dosen yang berisi informasi tentang dosen yang aktif mengajar dalam Universitas XYZ yang nantinya dapat digunakan sebagai sarana untuk memberitahukan informasi dari layanan karena tabel dosen berisi juga telepon

yang dapat dihubungi. Untuk lebih jelasnya dapat dilihat pada tabel 4.

Berdasarkan tabel normal kesatu salah satu tabel bentuk normal kedua adalah tabel ruang yang berisi ruang kelas mana saja yang dapat dipakai sebagai ruang aktif yang digunakan sebagai perkuliahan. Pada tabel 5.

Tabel 4. Dosen

NID	Nama Dosen	Binaan Prodi	Jenis Kelamin	Alamat	Telepon
L0110	Lukman Hakim	TI	L	Jalan Lodan no.12	089477368
L0999	Teady Matius	TI	L	Jalan Pluit utara no.23	098375637
L1153	Halim agung	TI	L	Citra 2	087777397
L0938	Asri	TI	P	Bekasi	089222880
L1078	Francka	SI	L	Ancol	021788859
L1203	Sherly Everlin	DK	P	Sunter	021657770

Tabel 5. Ruang

Id Ruang	Ruang
301	R301
302	R302

Berdasarkan tabel normal kesatu, dipecah menjadi salah satu tabel bentuk normal kedua adalah tabel jam yang berisi jam - jam perkuliahan yang ada. Tabel 6. Tabel jam berisi informasi sesi atau id_jam

Tabel 6. Jam

Id_jam	Jam Aktif
1	07.30 – 09.10
2	09.30 – 11.10

Berdasarkan tabel normal kesatu dipecah menjadialah satu tabel bentuk normal kedua yaitu tabel 7. tabel mata kuliah yang berisi mata kuliah yang ada dalam pengajaran yang ada dalam perkuliahan.

Tabel 7. Mata Kuliah

KMK	Mata Kuliah	SKS
TID6	Perancangan Web	4
TIO3	Pemograman Java	4

Tabel Normal ketiga (3NF)

Tabel 8. Karyawan

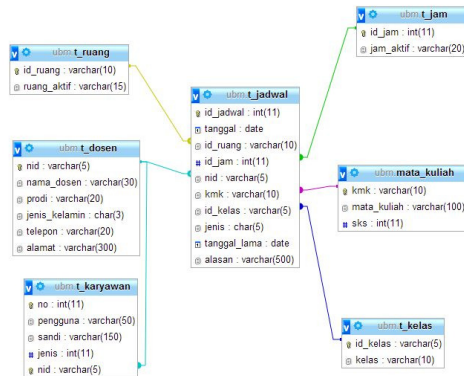
No	Pengguna	Sandi	NID	Jenis
1	Admin	admin	-	1
2	L0999	123456	L0999	2

Tabel 9. Jadwal

id	Tanggal	Id Ruang	Id jam	NID	K M K	ID Kel as	J e n i s	Tgl Perkul iahan	Alasan
1	28/03/2014	301	1	L0999	TI D6	IPT II	1		
2	28/03/2014	302	2	L0999	TI D6	IPT II	1		

Perancangan ERD

Untuk rancangan ERD ini sebagai Dummy yang ada pada Universitas XYZ untuk aplikasi penentuan kuliah pengganti.



Gambar 2. Rancangan ERD

Perancangan Algoritma B-Tree

Algoritma untuk operasi search, create, dan insert ditunjukkan di bawah. Perlu diperhatikan bahwa algoritma-algoritma ini hanya sekali pass. Dengan kata lain, algoritma tersebut tidak menelusuri pohon kembali. Karena B-tree berusaha melakukan akses disk seminimum

mungkin dan simpul-simpul biasanya disimpan pada disk, pass sekali ini akan mengurangi jumlah kunjungan pada simpul dan jumlah akses disk. Dua kali pass yang lebih sederhana pada pohon untuk mempesbaiki pelanggaran mungkin terjadi.

Karena semua simpul diasumsikan disimpan pada tempat penyimpanan sekunder (*disk*) daripada tempat penyimpanan primer (*memori*), semua referensi pada simpul yang diberikan sebelumnya dengan melakukan operasi *read* dilambangkan dengan *Disk-Read*. Sama halnya, apabila sebuah simpul dimodifikasi dan tidak lagi diperlukan, simpul harus ditulis pada tempat penyimpanan sekunder dengan operasi *write* dilambangkan dengan *Disk-Write*. Algoritma-algoritma di bawah diasumsikan bahwa semua simpul yang memiliki parameter telah memiliki operasi *Disk-Read* yang sesuai. Simpul baru dibuat dan di-assign dengan fungsi *Allocate-Node*. Implementasi secara detail dari *Disk-Read*, *Disk-Write*, dan *Allocate-Node* tergantung pada sistem operasi (*operating system*).

```
(InnoDB engine; index on
(nid, kmk, id_kelas));
Algoritma pencarian ( x adalah node
penunjuk pada cabang ) BTree-MySQL-Search
(x=null, nid='', kmk='', id_kelas='')
i=1;
while ( i < n[x] and (nid, kmk, id_kelas) >
keyi[x] )
i = i+1;
if ( i = n[x] and (nid, kmk, id_kelas) =
keyi[x] ) then
return keyi[x] -> rows;
else if ( leaf[x] ) then
return null;
else
Disk-Read(ci[x]);
Return BTree-MySQL-Search(ci[x],
nid, kmk, id_kelas);
```

Gambar 3. Proses Search

Operasi *Search* pada B-tree mirip dengan search pada *binary tree*. Dimulai dari akar, pohon ditelusuri dari atas sampai ke bawah.

Gambar 3 menunjukkan *pseudocodes* proses *search* pada B-tree Cari jadwal perkuliahan dimana *nid*='L0999' dan *kmk*='TID6' dan *id_kelas*='IPT11';

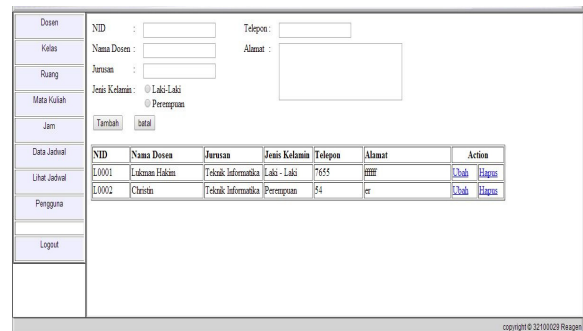
```

CREATE PROCEDURE Jadwal_hari(in
tanggal VARCHAR(15), in ruang_kelas
VARCHAR(10), in jam VARCHAR(10), in
id_dosen VARCHAR(10), matakuliah
VARCHAR(10), in kelas VARCHAR(10), in
bulan INT
begin
select WEEKDAY(t.tanggal) as id_hari,
t.id_jadwal as id_jadwal,
t.tanggal as tanggal,
r.ruang_aktif as ruang_aktif,
tjam.jam_aktif as jam_aktif,
d.nama_dosen as nama_dosen,
mk.kmk as kodemk,
mk.mata_kuliah as
mata_kuliah,
mk.sks as sks,
k.kelas as kelas_aktif,
t.jenis as jenis_jadwal,
t.tanggal_lama as
tanggal_lama,
t.alasan as alasan,
t.jenis as jenis
from t_jadwal t
left join t_jam tjam on
t.id_jam=tjam.id_jam
left join t_kelas k on
t.id_kelas=k.id_kelas
left join t_ruang r on
t.id_ruang=r.id_ruang
left join t_dosen d on t.nid=d.nid
left join mata_kuliah mk on
t.kmk=mk.kmk
where case when tanggal =" then " else
t.tanggal end = tanggal
and case when ruang_kelas =" then "
else t.id_ruang end = ruang_kelas
and case when jam =" then " else
t.id_jam end = jam
and case when id_dosen =" then " else
t.nid end =id_dosen
and case when matakuliah =" then "
else t.kmk end = matakuliah
and case when kelas =" then " else
t.id_kelas end =kelas
and case when bulan =" then " else
month(t.tanggal) end =bulan
or (t.nid is null
and t.kmk is null
and t.id_kelas is null
and case when bulan =" then " else
month(t.tanggal) end =bulan);
end
    
```

Gambar 4. Prosedur pencarian kelas pengganti

Pseudocodes proses *search* yang pada gambar 3 mencari dengan ketentuan `nid='L0001'` and `kmk='TID6'` and `id_kelas='1PT11'`. Saat pada bagian `'while (i < n[x] and (nid,kmk,id_kelas) > keyi[x])'` ini sebagai perulangan dalam pencarian dimana `n[x]` akan selalu bertambah untuk mencari key lainnya, ketika dibagian `'if (i = n[x] and (nid,kmk,id_kelas) = keyi[x]) then'` ini sebagai patokan pencarian dimana jika key yang ditentukan sama atau sesuai dengan ketentuan yang diberikan maka akan dijalankan `'return keyi[x] -> rows;'` yang akan disimpan sementara dalam rows jika tidak sesuai maka akan keluar dan akan diulang terus sampai seluruh cabang telah diperiksa.

Sistem pencarian kelas pengganti memiliki syarat yang harus dipenuhi sebelum digunakan yaitu jadwal sudah harus dibuat dahulu. Dalam pencarian kelas pengganti ini sistem menggunakan prosedur seperti pada gambar 4.



Gambar 5. Halaman Data Dosen

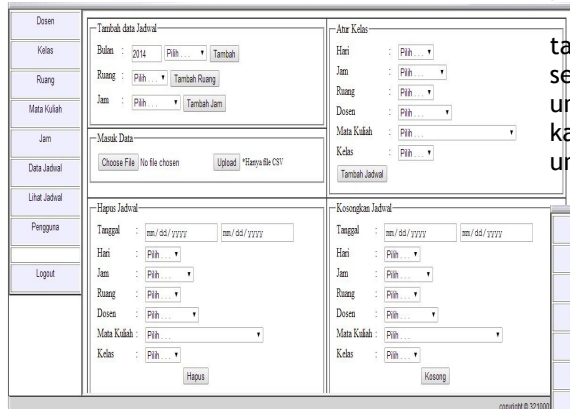
Tampilan Halaman Data Dosen

Pada Gambar 5 ini menampilkan halaman dosen yang berfungsi sebagai untuk melihat data - data dosen serta menambahkan dosen dosen baru. Pada kolom - kolom yang berada diatas harus diisi semua untuk menambahkan nama dosen yang baru dan menekan tombol tambah dan secara sendirinya tabel pada bagian bawah akan menampilkan nama dosen yang baru. Pada bagian tabel terdapat tombol ubah dan tombol hapus jika

menekan tombol hapus maka dengan sendirinya akan menghapus data dosen yang dipilih dan jika menekan tombol ubah berfungsi sebagai sarana untuk mengubah data dosen apabila terdapat perubahan baru atau salah memasukan data.

atau kelas pengganti. Halaman ini dapat diakses oleh admin dan dosen perbedaannya adalah pada bagian patokan nama dosen untuk admin pada bagian nama dosen dapat dipilih jika penggunaanya dosen maka bagian nama dosen akan secara dengan sendirinya akan diisi dan tidak dapat diubah untuk pencariannya.

Pada bagian bawah terdapat tampilan jadwal perkuliahan dosen serta ruang kelas yang dapat diambil untuk kelas pengganti. Pada bagian kanan dari tabel terdapat tombol ambil untuk mengambil kelas pengganti.

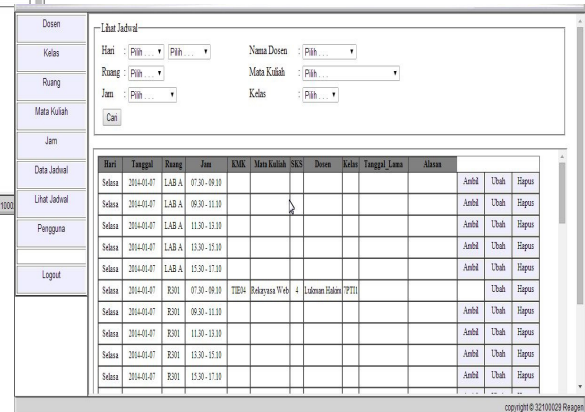


Gambar 6. Halaman Jadwal

Pada Gambar 6 ini menampilkan halaman data jadwal yang berfungsi sebagai untuk membuat jadwal perkuliahan dari data - data utama yang telah dibuat atau dimasukkan. Pada kolom tambah dan jadwal terdapat bulan, ruang, dan jam dimana kolom bulan berfungsi sebagai memilih bulan mana saja yang aktif dalam perkuliahan, sedangkan ruang dan jam berfungsi sebagai menambahkan jam atau ruang yang baru.

Pada kolom atur jadwal berfungsi untuk menentukan jadwal dalam perkuliahan yang harus diisi terlebih dahulu sebelum mencari jadwal perkuliahan pengganti. Pada kolom hapus data berfungsi sebagai menghapus data jadwal hari yang tidak perlu dipakai.

Pada Gambar 7 menampilkan halaman lihat jadwal yang berfungsi sebagai untuk melihat jadwal perkuliahan serta pengambilan kelas pengganti jika terdapat ruang yang dapat dipakai. Pada kolom atas terdapat patokan - patokan pencarian yang dapat diisi untuk mencari jadwal



Gambar 7. Halaman Lihat Jadwal

SIMPULAN

- Berdasarkan hasil pengujian yang dilakukan Sistem Penentuan Kelas Pengganti Untuk Dosen di Universitas XYZ Berbasis Web Dengan Metode B-Tree dapat membantu dalam menentukan kelas pengganti lebih mudah.
- Implementasi Algoritma B-Tree pada saat diujikan dapat melakukan pencarian data kuliah pengantinya, serta melacak kelas yang bentrok pada saat adanya kuliah pengganti.
- Sistem Penentuan Kelas Pengganti di Universitas XYZ Berbasis Web dilakukan secara online sehingga mengurangi terjadinya bentrok terhadap kelas pengganti yang telah diambil sebelumnya.

DAFTAR PUSTAKA

- [1] Delina, Hartini, et al.(2009). Database dengan SQL Server 2005. Mitra Wacana Media : Jakarta
- [2] Nugroho ,Bunafit . (2004). PHP dan MySQL dengan Editor Dreamwaver MX .ANDI : Yogyakarta
- [3] Raharjo,Budi., Heryanto, Imam., RK,Ejang. (2010). Modul Pemrograman WEB. Modula, Bandung
- [4] Utami, Ema.(2008). RDMS MENGGUNAKAN MS SQL SERVER 2000. Graha Ilmu:Yogyakarta
- [5] Peranginangin, Kasiman.(2006). Aplikasi WEB dengan PHP dan MYSQL. ANDI:Yogyakarta
- [6] Ladjamudin, Al Bahra bin.(2013).Analisi dan Desain Sistem Infrmasi.Graha Ilmu : Tangerang
- [7] Ratri, Sindy Gita. (2006). <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2006-2007/Makalah/Makalah0607-49.pdf> . Tanggal akses : 10 Mei 2014 pukul 20:00 WIB