**LAPORAN TUGAS BESAR**

**STRUKTUR DATA**

**(Data Artikel Ilmiah)**

**Laporan**

Diajukan untuk memenuhi tugas pada mata kuliah

Struktur Data

**Oleh:**

Fendi Irfan Amorokhman (1301191447)



**PROGRAM STUDI S-1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**UNIVERSITAS TELKOM**

**BANDUNG 2020**

**LAPORAN TUGAS BESAR**

**STRUKTUR DATA**

**(Data Artikel Ilmiah)**

Fendi Irfan Amorokhman

[fendiirfan@student.telkomuniversity.ac.id](mailto:fendiirfan@student.telkomuniversity.ac.id)

**BAB 1**

1.Latar Belakang

Dewasa ini banyak pekerjaan atau tantangan yang muncul akibat banyaknya kebutuhan yang ada. Perkembangan zaman yang semakin pesat juga menuntut kita untuk melakukan semuanya dengan efisien , penggunaan teknologi mutlak untuk menopang kegiatan pengefisienan tersebut. Salah Satu teknologi yang bisa gunakan dalam melakukan hal terbut ialah software komputer. Dalam software kita mengenal yang namanya program komputer untuk menjalankan perintah perintah komputasi yang telah ditentukan. Kita juga telah mengetahui yang anmanya data, penggunaan data pada program akan menambah kegunaan dari program tersebut. Saya tertarik untuk memanfaatkan imu saya untuk melakukan pngembangan program untuk menjadi *platform* atau wadah pusat informasi perpustakaan.

2. Rumusan masalah

* Bagaimana program tersebut melakukan pengumpulkan data?
* Bagaimana program tersebut memberikan informasi terhadap pengguna dari data ata yang telah dikumpulkan?
* Apa saja fitur yang tersedia dalam program tersebut?

3. Tujuan

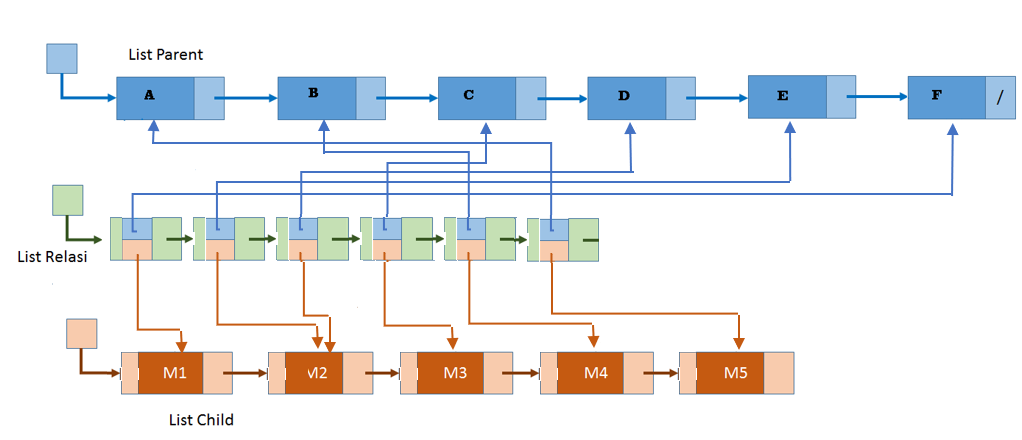
* Meng-efisiensikan pengumpulan data Artikel.
* Memudakan pencarian suatu Artikel.
* Memberikan informasi mengenai jenis keyword yang paling banyak atau paling sedikit memiliki judul artikel ilmiah.

4. Pembagian Tugas

Fendi Irfan Amorokhman merupakan pengembang dari program data artikel ilmiah pada laporan ini.  
5. Video presentasi (link video youtube)

BAB 2

1. Rancangan List



1. Pada list berwarna biru adalah list parent, yang memuat judul artikel ilmiah, penulis artikel ilmiah, penerbit artikel ilmiah tersebut, tahun diterbitkan, abstrak dari artikel ilmiah. List ini bermodelkan single linked list circular.
2. Pada list yang berwarna hijau adalah list relasi, yang menghubungkan keterhubungan antara element di list *parent* dan list *child.* List ini bermodelkan singular double linked list.
3. Pada list berwarna cokelat adalah list anak, yang memuat data *keyword* suatu atau jenis artikel ilmiah .list ini bermodelkan singular double linked list .
4. Fungsi dan prosedur

Fungsi atau Prosedur list parent

|  |  |
| --- | --- |
| Fungsi / prosedur | keterangan |
| void createListParent(Parent &p);  I.S. -  FS. Terdefinisi List | Membuat list |
| void insertFirstParent(Parent &p,addressParent a);  IS. TERDEFINISI LIST MUNGKIN KOSONG DAN ADDRES PARENT\_H\_INCLUDED  FS. ADDRES PARENT DIMASUKAN DIAWAL LIST | Prosedur ini menaruh element yang ditunjuk oleh addressparent untuk dimasukkan ke dalam list parent pada awal list. |
| void createElementParent(addressParent &p,string judul,string penulis,string publisher,string tahun,string abstrak);  IS. -  FS. TERDEFINISI ELEMENT DENGAN INFO BERISI JUDUL... | Membentuk element baru yang memuat judul artikel ilmiah, penulis artikel ilmiah, penerbit artikel ilmiah tersebut, tahun diterbitkan, abstrak dari artikel ilmiah. |
| addressParent findParent(Parent p,string judul);  IS. TERDEFINISI LIST PARENT MUNGKIN KOSONG DAN SEBUAT JUDUL  FS. TERDEFINISI ADDRESS PARENT YANG BERADA DI LIST PARENT DENGAN STRING JUDUL DAN JIKA TIDAK DITEMUKAN ATAU KOOSNG RETURN NULL | Fungsi ini bertujuan mencari alamat element dengan info judul lalu mengembalikannya dan jika tidak ada mengembalikan NULL. |
| Void deleteElementParent(Parent &p,addressParent x);  IS. TERDEFINISI LIST PARENT TIDAK KOSONG DAN ADDRES PARENT YANG INGIN DIHAPUS  FS. ELEMEN ADDRESPARENT TERHAPUS DI LIST | Prosedur ini berfungsi menghapus element yang di tunjuk oleh addressParent |
| void tampilkanSemuaJudulartikel(Parent p,addressParent Ap);  I.S : -  F.S : : Menampilkan semua judul Artikel | Prosedure ini berfungsi mengolah perintah menampilkan semua judul artikel |

Fungsi / prosedur list relasi

|  |  |
| --- | --- |
| Fungsi/ prosedur | keterangan |
| void createList(List\_relasi &L);  I.S : list relasi belum terbentuk  F.S : List relasi sudah terbentuk |  |
| void insertFirst(List\_relasi &L, address\_relasi P);  I.S : list mungkin kosong, P sudah dialokasikan  F.S : menempatkan elemen P sebagai elemen pertama pada list relasi |  |
| void insertLast(List\_relasi &L, address\_relasi P);  I.S : list mungkin kosong, P sudah dialokasikan  F.S : menempatkan elemen P sebagai elemen terakhir pada list relasi |  |
| void insertAfter(address\_relasi Prec, address\_relasi P);  I.S : list mungkin kosong, P dan prec alamat salah satu elemen list  F.S : menempatkan elemen beralamatkan p sesudah elemen beralamat prec pada list relasi |  |
| void deleteFirst(List\_relasi &L, address\_relasi &P);  I.S : List tidak kosong  F.S : P adalah alamat elemen pertama list sebelum elemen pertama list di hapus, elemen pertama list hilang dan list mungkin menjadi kosong. |  |
| void deleteLast(List\_parent &L, address\_parent &P);  I.S : list tidak kosong  F.S : : P adalah alamat elemen terakhir list sebelum elemen pertama list di hapus, elemen terakhir list hilang dan list mungkin menjadi kosong. |  |
| void deleteAfter(List\_relasi &L, address\_relasi Prec, address\_relasi &P);  Prec, address\_parent &P);  I.S : list tidak kosong , prec alamat salah satu elemen list  F.S: P adalah alamat dari next(prec) , menghapus next prec dari list . |  |
| address\_relasi alokasi(address\_parent P, address\_child C);  / mengirimkan address dari alokasi sebuah elemen relasi jika alokasi berhasil, dan nilai address tidak NULL dan jika gagal , maka sebaliknya. |  |
| void printAllInfo(List\_relasi L);  I.S : list mungkin kosong  FS : jika list tidak kosong , akan menampilkan semua info yang ada di list relasi |  |
| void printInfoAnak(List\_relasi L, string ayah, string ibu);  I.S : list mungkin kosong  FS : jika list tidak kosong , akan menampilkan relasi anak di list relasi |  |
| void printInfoInduk(List\_relasi L, string x);  I.S : list mungkin kosong  FS : jika list tidak kosong , akan menampilkan relasi orang tua di list relasi |  |
| void printIndukMinMax(List\_relasi L);  I.S : list mungkin kosong  FS : jika list tidak kosong , akan menampilkan jumlah relasi terbesar atau terkecil pada list relasi |  |
| void deleteAnak(List\_parent K, List\_relasi L, List\_child M, string anak); |  |

Fungsi/ prosedur list child

|  |  |
| --- | --- |
|  |  |
| void createList(List\_child &L); | Prosedur ini berguna untuk meng-inisialisasikan dalam membuat list anak |
| void insertFirst(List\_child &L, address\_child P); | Prosedur ini berguna untuk memasukan atau menambah elemen baru, sebagai elemen pertama pada list anak |
| void insertLast(List\_child &L, address\_child P); | Prosedur ini berguna untuk memasukan atau menambah elemen baru, sebagai elemen terakhir pada list anak |
| void insertAfter(address\_child Prec, address\_child P); | Prosedur ini berguna untuk memasukan atau menambah elemen baru, elemen baru akan di tambahkan setelah alamat prec pada list anak |
| void deleteFirst(List\_child &L, address\_child &P); | Prosedur ini berguna untuk menghapus data anak pada elemen pertama list anak |
| void deleteLast(List\_child &L, address\_child &P); |  |
| void deleteAfter(List\_child &L, address\_child Prec, address\_child &P); |  |
| address\_child alokasiChild(string nama, string NIK, string peranan, string tempat, string tanggal, string agama, string pendidikan, string pekerjaan); |  |
| address\_child findElm(List\_child L, string anak); |  |
| void printInfo(List\_child L); |  |

3 .Pengujian

Bab 3  
kesimpulan dan saran