

LAPORAN TUGAS BESAR
IF2030/Algoritma dan Struktur Data

TRO LO LO LO LOL GRAM

Dipersiapkan oleh:

Kelompok 2

Arieh Rahman / 13511020

Azalea Fisitania / 13511028


Dinah Kamilah Ulfa / 13511087

Rifki Afina Putri / 13511066

Setyo Legowo / 13511071

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2030-TB-02-03</i>		<i>28</i>
		<i>Revisi</i>	<i>1</i>	<i>03/12/2012</i>

Daftar Isi

1	Ringkasan.....	3
2	Penjelasan Tambahan Spesifikasi Tugas	3
2.1	Priority Queue.....	3
2.2	Binary Search Tree	3
3	Struktur Data.....	3
3.1	List	4
3.2	List Rekursif.....	4
3.3	Pohon Biner	4
3.4	Priority Queue.....	4
3.5	Mesin Karakter.....	4
3.6	Mesin Kata	5
4	Program Utama	5
5	Algoritma-Algoritma Menarik.....	7
5.1	Dirent.h	8
5.2	CheckFileFolder(char * name, char dOf)	8
6	Data Test	9
6.1	Modus Penulisan	9
6.2	Modus Pengiriman	10
6.3	Modus Penerimaan.....	10
6.4	Modus Pengaturan	11
7	Test Script	13
8	Pembagian Kerja dalam Kelompok	18
9	Lampiran	19
9.1	Deskripsi Tugas Besar	19
9.2	Notulen Rapat	26
9.3	Log Activity Anggota Kelompok	26
9.4	Asistensi.....	29

1 Ringkasan

Tugas besar mata kuliah Algoritma Struktur Data IF2030 kali ini adalah membuat sebuah program berbasis sistem telegram untuk menulis, mengirim, menerima, dan membaca pesan. Secara umum, program menyediakan fitur bagi user untuk menulis pesan (mengetikkan masukan berupa string) pada program. Secara internal, pesan tersebut diencode ke bahasa TELEGRAM dan disimpan dalam file eksternal di folder OUTBOX. Setelah itu, user bisa mengirim pesan yang telah ditulis. Program akan membuat TELEGRAM diencode ke kode MORSE dan disimpan dalam file eksternal di folder SENDING. Fitur selanjutnya adalah menerima pesan dan membaca pesan. Kode MORSE akan didecode ke TELEGRAM dan disimpan dalam file eksternal di folder INBOX. Secara internal, TELEGRAM didecode lagi ke dalam bentuk pesan yang aslinya dan ditampilkan pada user. Selain fitur yang telah disebutkan sebelumnya, ada pun statistik dan pengaturan yang berkaitan dengan program. Semua fitur tersebut dikelompokkan ke dalam empat Modus: Penulisan, Pengiriman, Penerimaan, dan Pengaturan. Sebagai penunjang program, dibutuhkan ADT (Abstract Data Type) yakni: list, list rekursif, mesin karakter, mesin kata, dan pohon biner.

Laporan ini berisi spesifikasi dari program telegram, struktur data yang digunakan pada program ini, cara kerja program utama, serta pembagian tugas, progress pengerjaan tugas, serta log activity dari setiap anggota kelompok.

Sebagai kesimpulan, program sistem telegram yang penulis buat telah memenuhi semua spesifikasi yang diwajibkan dalam tugas, dengan beberapa fitur tambahan dan beberapa bonus yang dikerjakan.

2 Penjelasan Tambahan Spesifikasi Tugas

Dalam pengerjaan tugas besar ini, penulis memasukkan beberapa fitur tambahan, diantaranya priority queue dan binary search tree.

2.1 Priority Queue

Struktur data priority queue digunakan pada modus pengiriman saat mentransmit semua pesan ke dalam folder inbox. Pesan-pesan yang akan dikirimkan diurutkan berdasarkan prioritasnya. Prioritas merupakan elemen dari tipe bentukan ElmtQueue yang bertipe integer. Angka 1 menunjukkan prioritas kritikal, 2 menunjukkan prioritas penting, 3 menunjukkan prioritas segera, dan 4 menunjukkan prioritas biasa. Pesan dengan angka paling kecil atau prioritas paling tinggi akan ditransmit terlebih dahulu.

2.2 Binary Search Tree

Struktur data binary search tree digunakan untuk memudahkan proses encoding dan decoding dari morse ke bahasa biasa.

3 Struktur Data

Berikut ini adalah beberapa struktur data yang digunakan untuk mengatasi persoalan-persoalan pada pengerjaan tugas besar ini:

STEI- ITB	IF2030-TB-02-03	Halaman 3 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

3.1 List

Struktur data list hampir digunakan pada seluruh modul dalam sistem telegram ini. Pada modul penulisan, struktur data list digunakan untuk menyimpan daftar pelanggan dan untuk menyimpan dan mengupdate data statistik pesan. Pada modul pengiriman dan penerimaan struktur data list juga digunakan untuk mengupdate data statistik pesan. Pada modul konfigurasi, list dipakai untuk mengupdate list pelanggan. Hampir semua data yang disimpan dalam file eksternal, diassign ke dalam sebuah list saat akan diproses dalam program. Struktur data list ini diimplementasikan dalam ADT list dengan nama file list.c dan list.h.

3.2 List Rekursif

Struktur data list rekursif ini dibuat sebagai syarat agar dapat membuat ADT pohon biner. List rekursif ini diimplementasikan dalam ADT list rekursif dengan nama list_rekursif.c dan list_rekursif.h.

3.3 Pohon Biner

Struktur data pohon biner digunakan untuk proses encoding dan decoding dari bahasa morse ke bahasa telegram. Struktur data pohon biner dapat mempermudah proses decoding dan encoding tersebut. Struktur data pohon biner diimplementasikan sebagai ADT pohon biner dengan nama file pohon_biner.c dan pohon_biner.h.

3.4 Priority Queue

Struktur data priority queue digunakan saat pengiriman transmit pesan ke folder inbox. Antrian dari pengiriman pesan ini akan diurutkan berdasarkan waktu dan prioritasnya. Pesan dengan prioritas yang lebih tinggi akan masuk pada bagian antrian yang lebih awal. Modul pada sistem telegram yang menggunakan struktur data ini adalah modul pengiriman. Struktur data priority queue diimplementasikan dalam bentuk ADT priority queue dengan nama file p_queue.c dan p_queue.h.

3.5 Mesin Karakter

Bagian dari program yang membutuhkan mesin karakter adalah encoder-decoder telegram ke morse serta encoder-decoder morse ke telegram. ADT Mesin karakter dipilih karena translasi pesan dalam bentuk telegram ke morse dan sebaliknya memanfaatkan file eksternal. String tidak disimpan dalam array of character, melainkan dalam pita karakter yang direkam dalam file eksternal .txt (ketentuan dari Tugas Besar). Penyimpanan hasil translasi dalam file eksternal .txt sendiri sebenarnya memudahkan pemrogram untuk mengecek apakah ada kesalahan dalam hasil translasi dibandingkan dengan mengakses indeksnya satu-satu seperti dalam array of character.

Mesin karakter di sini bertugas untuk mengakuisi per alfabet dalam satu pita karakter untuk selanjutnya ditranslasi ke dalam kode morse. Translasi satu huruf alfabet ke dalam kode morse menggunakan ADT Pohon: Binary Search Tree.

Struktur data mesin karakter dibuat sebagai syarat agar dapat membuat ADT mesin kata. Karena pada dasarnya kata adalah gabungan dari beberapa karakter. Mesin karakter ini diimplementasikan dalam ADT mesin karakter dengan nama mesinkar.c dan mesinkar.h.

3.6 Mesin Kata

Struktur data mesin kata digunakan pada semua modus dalam sistem telegram ini. Pada modus penulisan, mesin kata digunakan saat penulisan pesan. Masukan user direkam dalam mesin kata, kemudian disimpan di dalam folder outbox. Pada modus pengiriman, mesin kata digunakan saat proses transmit pesan, khususnya saat proses encode dan decode morse. Kata-kata dalam bahasa telegram diubah menjadi bahasa morse dan direkam dengan menggunakan mesin kata. Pada modus penerimaan, mesin kata digunakan untuk membaca dan menghapus pesan. Pada modus konfigurasi, mesin kata digunakan saat membaca dan mengupdate list item yang ada pada kamus. Struktur data mesin kata ini kebanyakan digunakan untuk memproses suatu data yang berhubungan dengan file eksternal. Karena untuk membaca, menulis, dan memproses suatu file eksternal akan lebih mudah jika menggunakan struktur data mesin kata. Mesin kata ini diimplementasikan dalam ADT mesin kata dengan nama file mesinKata_v1.c dan mesinKata_v1.h.

4 Program Utama

Header-header yang dipakai dalam program utama adalah header-header standar seperti `stdio.h`, `stdlib.h`, `boolean.h`, juga header-header tambahan yang berkaitan dengan spesifikasi sistem telegram seperti `sys/stat.h` (untuk mengecek ada tidaknya suatu file dalam folder), `time.h` (untuk mengcapture waktu), `list.h` (struktur data yang digunakan khususnya untuk data-data dalam file eksternal yang diassign ke dalam suatu list), `p_queue.h` (priority queue, telah dijelaskan pada bagian sebelumnya), `mesinkar.h` (mesin karakter), `interface.h` (untuk tampilan interface yang lebih menarik), `konfigurasi.h`, `modusKonfigurasi.h`, `ModusPenerimaan.h`, `ModusPengiriman.h`, serta `penulisan.h`.

Cara kerja program utama dimulai pada bagian inisialisasi. Jika program tidak menemukan file dan atau folder yang dibutuhkan program maka harus dibuat dan program direset ke kedalam fresh install (default configuration). Lalu program menyiapkan beberapa hal, yaitu membuat list karakter khusus, membuat list daftar harga, membuat list daftar pelanggan, membuat list file pesan di Inbox, Sending, dan Outbox, menjalankan proses pengiriman dari folder Sending ke folder Inbox, dan membuat pita karakter pohon biner.

Setelah tahap inisialisasi selesai, maka tahapan selanjutnya adalah tampilan menu untuk memilih modus yang akan digunakan.

A. Menu untuk Modus Penulisan

1. Membuat pesan (Kirim pesan)

- Cek nama pengguna pengirim lalu penerima apakah sudah terdaftar atau belum.
- Pesan harus dapat direpresentasikan dengan semua karakter (abjad, tanda baca, angka, dan karakter khusus) dapat diubah ke abjad-abjad agar dapat dikirimkan melalui telegram/kode morse. Maka selalu ada pengecekan semua karakter yang dimasukkan pengguna sebelum melanjutkan ke proses selanjutnya.
- Jika kedua hal diatas tidak dipenuhi maka mengulang masukan dengan maksimum mengulang 3 kali kecuali saat nama pelanggan yang terdaftar menulis di bagian pengirim dan penerima itu sama maka program tidak akan mengirim pesan tersebut dan langsung kembali ke modus Penulisan. Jika sudah 3 kali belum berhasil maka sistem batal mengirimkan pesan dan keluar dari modus Penulisan.

- Semua pesan disimpan ke dalam suatu file. Pesan yang terdiri dari abjad disimpan ke dalam folder Outbox.
 - Nama file pesan memiliki ketentuan "YYMMDDHHMMSS-<pengirim>.txt" dimana <pengirim> adalah nama pengirim, YY adalah 2 bilangan terakhir dari tahun, MM adalah bulan pengiriman, DD adalah tanggal pengiriman, HH adalah jam pengiriman, MM adalah menit pengiriman, dan SS adalah detik pengiriman.
 - Setelah proses pengiriman diatas selesai ditampilkan ke layar jumlah kata dari masing-masing jenis kata dan biaya.
2. Menghitung Harga
 - Mencetak informasi perintah pesan terakhir yang dikirim. Informasi itu meliputi: pengirim, penerima, perhitungan harga dan jumlah kata dari masing-masing jenis kata dari pesan terakhir yang dikirim.
 - Jika program baru dijalankan maka informasi tentang menghitung harga akan menampilkan dengan angka 0 yang berarti belum ada pesan yang dikirim sejak program dijalankan.
 3. Statistik
 - Statistik pada modul ini meliputi: Nama pengirim, Jumlah Pesan yang dikirim, Rata-rata jumlah total kata dalam sebuah pesan, dan total harga dari pesan-pesan yang pernah dikirim.
 4. Ubah Modus Penggunaan
 - Pindah ke menu utama
 5. Keluar
 - Keluar dari program
- B. Menu untuk modul pengiriman
1. Transmit Semua Pesan
 - Membaca semua file pengiriman yang ada di Outbox dan mentranslasikan ke file lain yang berisi sandi morse dan meletakkannya pada folder Sending dengan nama file yang sama. Setelah itu, file pesan yang ada di folder Outbox dihapus.
 - Setiap program di jalankan atau menu dipilih atau saat program sedang berjalan maka pesan-pesan pada folder Sending ditranslasikan ke folder Inbox dengan nama file yang sama diantrikan dengan selang waktu 30 detik baru ditransfer. Antriannya berdasarkan prioritas pesan.
 2. Statistik
 - Mencetak statistik jumlah pengiriman pesan pada tiap jamnya.
 3. Ubah Modus Penggunaan
 - Pindah ke menu utama
 4. Keluar
 - Keluar dari program
- C. Modus Penerimaan
1. List Pesan
 - Menampilkan nama-file yang telah dikirim (pesan-pesan yang ada di Inbox).
 - Menampilkan ID setiap file yang akan dibuka.
 2. Baca Pesan
 - Lalu untuk membuka file pesan, pengguna harus memasukkan ID yang ditampilkan di submenu List Pesan.

- Pesan yang disampaikan adalah pesan hasil konversi dari representasi alfabetis dan numerik ke pesan yang mudah dibaca pengguna.
- 3. Hapus Pesan
 - Untuk menghapus pesan, pengguna harus memasukkan ID nama file yang akan dihapus di submenu List Pesan.
 - Pesan yang dihapus berarti file pesan yang ada di Inbox dihapus.
- 4. Statistik
 - Statistik pada modus ini meliputi: Nama Penerima, Jumlah Pesan yang diterima dan Jumlah pesan yang dibaca.
- 5. Ubah Modus Penggunaan
 - Pindah ke menu utama
- 6. Keluar
 - Keluar dari program
- D. Modus Pengaturan
 1. Daftar Harga
 - Menampilkan list harga dari jenis-jenis kata.
 - Jika program pertama kali dijalankan, maka harga akan diset secara default.
 - Harga dapat diatur dengan edit di file konfigurasi langsung.
 2. List Item Kamus
 - Mencetak ke layar daftar item yang telah terdaftar.
 - Jika program pertama kali dijalankan, maka list item (karakter khusus) akan diset secara default.
 - List item kamus dapat diatur dengan edit di file konfigurasi.
 - Maksimum karakter yang dapat ditampung program adalah 30 karakter.
 3. Penambahan Item Kamus
 - Menambah item baru ke dalam kamus, jika item tersebut belum terdefinisi ke dalam kamus.
 4. Penghapusan Item Kamus
 - Menghapus item yang telah terdefinisi pada kamus.
 5. List Pelanggan
 - Mencetak ke layar list pelanggan yang telah terdaftar di sistem.
 6. Penambahan Pelanggan
 - Mendaftarkan pelanggan baru ke dalam sistem, jika pelanggan belum terdaftar dalam sistem.
 7. Penghapusan Pelanggan
 - Menghapus entri pelanggan yang terdefinisi dalam sistem.
 8. Ubah Modus Penggunaan
 - Pindah ke menu utama
 9. Keluar
 - Keluar dari program

5 Algoritma-Algoritma Menarik

Dalam pengerjaan tugas besar ini, terdapat beberapa algoritma yang menarik, yaitu sebagai berikut:

STEI- ITB	<i>IF2030-TB-02-03</i>	Halaman 7 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

6.5 *Dirent.h*

Algoritma ini ada pada hampir seluruh bagian modul penerimaan, kecuali pada bagian statistik. Berikut ini beberapa potongan isi algoritma ini:

```
#include <dirent.h>

void TampilListPesan (void)
/* I.S : List pelanggan dan kamus pada modul pengaturan telah
terdefinisi.
   F.S : Menampilkan list pesan-pesan yang telah terkirim pada
folder inbox. */
{
    // Kamus Lokal
    int i;
    DIR *dir;
    struct dirent *dirent;
    struct stat st;
    char ifile[100];

    // Algoritma
    dir = opendir("./Inbox");
    i = 0;
    while ((dirent = readdir(dir)) != NULL) {
        snprintf(ifile,100,"Inbox/%s",dirent->d_name);
        if((stat(ifile,&st) == 0) && (ifile[6] != '.')) {
            printf("  %d. %s",i+1,ifile);
            printf("\n");
            i++;
        }
    }
    closedir(dir);
    if(i == 0)
        printf("  Pesan tidak ada (kosong).\n");
}
```

Dengan menginclude *dirent.h* pada header, maka program dapat membuka suatu direktori dengan fungsi *opendir()*, kemudian membaca seluruh nama file yang ada pada direktori tersebut dengan fungsi *readdir()*. Fungsi ini dapat membuat algoritma menjadi lebih mangkus, karena dapat membaca isi file tanpa menggunakan file eksternal. Sehingga memori yang terpakai pun tidak terlalu banyak karena tidak perlu digunakan file eksternal untuk membaca list isi pesan.

6.6 *CheckFileFolder(char * name, char dOf)*

Algoritma ini berada di *main_program.c*. Algoritma ini berguna saat file executable dieksekusi sendiri (pertama kali dijalankan dan file folder yang digunakan program ini belum

terbentuk) maka dapat membentuk folder dan file baku program ini. Program ini dapat berjalan jika menggunakan header <sys/stat.h>. Algoritmanya adalah sebagai berikut:

```
void CheckFileFolder(char * name, char dOf)
// { I.S. : name dan dOf terdefinisi }
// { F.S. : Terbentuk folder atau file kosong sesuai dengan
namanya jika
//          file yang ditunjuk tidak ditemukan }
// { Name adalah nama hierarki file atau folder dari program
tersebut berjalan }
// { Jika dOf = 'd' maka name adalah direktori, jika dOf = 'f'
maka name adalah
//      nama file. }
{
    struct stat st;
    FILE *fp;

    if(stat(name,&st) != 0)
    {
        printf(" %s tidak ditemukan. Create it...",name);
        if(dOf == 'd') {
            mkdir(name,S_IRWXU);
        } else if (dOf == 'f') {
            fp = fopen(name,"w");
            fprintf(fp,"%c",MARK);
            fclose(fp);
        }
        printf("\n");
    }
}
```

6 Data Test

6.1 Modus Penulisan

1. Kirim
 - a. Submenu Kirim digunakan untuk membuat pesan baru.
 - b. Hasil yang dapat didapatkan dengan submenu Kirim yaitu membentuk file pesan baru di folder Inbox dan mencatat data yang diperlukan untuk statistik penulisan.
 - c. Data Test yang dapat diberikan
Masukan:
Prioritas : 4
Nama Pengirim : AZALEA
Nama Penerima : RIFKI
Pesan :

STEI- ITB	IF2030-TB-02-03	Halaman 9 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

Data Test 1, karakter % dan @ dapat dimasukkan sebagai pesan

Prioritas : 1

Nama Pengirim : ARIEF

Nama Penerima : SETYO

Pesan :

Data Test 2, karakter & dan # dapat dimasukkan sebagai pesan

2. Hitung Harga

- Submenu Hitung Harga digunakan untuk melihat pesan terakhir yang dikirim saat program dijalankan.
- Jika program pertama kali dijalankan, informasi harga, jumlah Total kata, jumlah kata pendek, jumlah kata panjang, Tanda Baca, Lain-lain, dan ENTER akan menampilkan 0 (nol). Jika tidak maka akan ditampilkan biaya dan penggunaan karakter di pesan terakhir.

3. Statistik

- Submenu Statistik digunakan untuk mencetak statistik pesan yang ditulis oleh pelanggan.
- Hasil yang didapat dari fitur ini adalah nama pengirim, jumlah pesan yang dikirim, rata-rata jumlah total kata dalam sebuah pesan, dan total harga dari pesan yang dikirim.

6.2 Modus Pengiriman

1. Transmit Semua Pesan

- Submenu Transmit Semua Pesan digunakan untuk mengirimkan pesan.
- Pada fitur ini terjadi encode dan decode dari morse ke bahasa telegram. Pesan yang sudah di-encode ke bahasa morse akan disimpan dalam folder sending, kemudian pesan tersebut akan terhapus dalam folder outbox. Kemudian, pesan akan dikirimkan ke folder inbox setiap 30 detik. Secara internal program akan men-decode pesan kembali ke bahasa telegram.
- Jika semua pesan berhasil dikirim maka pada layar akan muncul:
[jumlah pesan yang dikirim] pesan telah dikirim.

2. Statistik

- Submenu Statistik digunakan untuk mencetak jumlah pengiriman pesan tiap jam.
- Hasil yang didapat adalah tanggal dan waktu, serta jumlah pesan yang dikirim pada waktu tersebut.

6.3 Modus Penerimaan

1. List Pesan

- Submenu List Pesan digunakan untuk menampilkan list pesan yang ada pada folder inbox.
- Hasil yang didapatkan pada fitur ini yaitu, daftar nama-nama list pesan yang ada:

1. Inbox/121203044604-SETYO.txt

STEI- ITB	IF2030-TB-02-03	Halaman 10 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

2. Inbox/121203053012-AZALEA.txt

2. Baca Pesan

- Submenu Baca Pesan digunakan untuk membaca pesan sesuai masukan ID.
- Hasil yang didapat adalah isi pesan yang telah diubah dari representasi alphabet (bahasa telegram) menjadi tanda baca biasa sehingga lebih mudah dibaca.
- Data test yang diberikan:

Masukkan id pesan : [ID sesuai dengan list ID yang ada]

NAMA PENGIRIM: SETYO

NAMA PENERIMA: ARIEF

PESAN: DATA TEST, TANDA BACA BERHASIL DIUBAH.

Masukkan id pesan : [masukan ID tidak sesuai]
ID pesan tidak ditemukan atau salah.

3. Hapus Pesan

- Submenu Hapus Pesan digunakan untuk menghapus pesan sesuai masukan ID.
- Hasil yang didapat adalah pesan dengan ID yang dimasukkan akan terhapus dari folder inbox.
- Data test yang diberikan:

Masukkan id pesan : [ID sesuai dengan list ID yang ada]

Pesan [Nama pesan] telah terhapus dari sistem.

Masukkan id pesan : [masukan ID tidak sesuai]
ID pesan tidak ditemukan atau salah.

4. Statistik

- Submenu Statistik digunakan untuk mencetak data statistik pesan.
- Hasil yang didapatkan pada fitur ini yaitu, keluaran nama pengirim, jumlah pesan yang diterima, dan jumlah pesan yang dibaca:

1. AZALEA 2/1 pesan

6.4 **Modus Pengaturan**

1. Daftar Harga

- Submenu Daftar Harga digunakan untuk menampilkan list harga dari jenis-jenis kata.
- Hasil yang diberikan pada submenu ini adalah daftar harga sesuai jenis kata, yaitu kata pendek seharga 500, kata panjang seharga 800, tanda baca seharga 200, lain-lain seharga 700, dan 'enter' seharga 0.

2. List Item Kamus

- Submenu List Item Kamus digunakan untuk menampilkan list item yang ada pada kamus.
- Hasil yang diberikan pada submenu ini adalah daftar tanda baca berikut namanya dalam bahasa telegram.

- ! - TANDASERU
- " - KUTIPGANDA

3. Penambahan Item Kamus

- a. Submenu Penambahan Item Kamus digunakan untuk menambahkan list item yang ada pada kamus.
- b. Hasil yang didapat pada submenu ini adalah bertambahnya daftar item kamus.
- c. Data test yang diberikan:

Masukkan karakter khusus yang akan ditambahkan: /
Warning: Karakter yang Anda masukkan terdefinisi di kamus atau termasuk karakter alphanumeric.

Masukkan karakter khusus yang akan ditambahkan: {
Masukkan kata untuk karakter khusus tersebut: KURUNGBUKAKURAWAL
Berhasil ditambahkan ke kamus.

4. Penghapusan Item Kamus

- a. Submenu Penghapusan Item Kamus digunakan untuk menghapus list item yang ada pada kamus.
- b. Hasil yang didapat pada submenu ini adalah berkurangnya daftar item kamus.
- c. Data test yang diberikan:

Masukkan karakter khusus yang akan ditambahkan: {
Proses penghapusan berhasil.

5. List Pelanggan

- a. Submenu List Pelanggan digunakan untuk menampilkan nama pelanggan yang telah terdaftar.
- b. Hasil yang didapatkan pada submenu ini adalah nama-nama daftar pelanggan, dan menampilkan pesan 'Daftar pelanggan kosong.' Apabila belum ada nama pelanggan yang terdaftar.

6. Penambahan Pelanggan

- a. Submenu Penambahan Pelanggan digunakan untuk menambahkan daftar pelanggan.
- b. Hasil yang didapat dari submenu ini adalah bertambahnya nama pelanggan.
- c. Data test yang diberikan:

Masukkan nama pengguna baru: rifki
Nama pengguna baru tersimpan dengan baik.

7. Penghapusan Pelanggan

- a. Submenu Penghapusan Pelanggan digunakan untuk menghapus nama pelanggan yang telah terdaftar.
- b. Hasil yang didapat dari submenu ini adalah berkurangnya nama pelanggan.
- c. Data test yang diberikan:

Masukkan nama pengguna yang akan dihapus: budi
Nama pengguna baru yang Anda masukkan tidak terdaftar.

Masukkan nama pengguna yang akan dihapus: rifki

STEI- ITB	IF2030-TB-02-03	Halaman 12 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

Nama pengguna RIFKI dihapus dari daftar.

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Kirim	Membuat pesan baru	<ol style="list-style-type: none"> 1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 1 untuk memilih modul Penulisan, lalu tekan ENTER 3) Ketik angka 1 untuk memilih Submenu Kirim 4) Masukkan angka prioritas, lalu tekan ENTER 5) Masukkan nama pelanggan pengirim, lalu tekan ENTER 6) Masukkan nama pelanggan penerima, lalu tekan ENTER 7) Masukkan pesan yang akan dikirim, lalu tekan ENTER 8) Jika nama pelanggan pengirim atau penerima atau karakter di dalam pesan terdapat karakter yang tidak terdaftar maka pengguna dapat mengulang sebagian formulir pesan. 9) Jika masukan nama pelanggan atau pesan telah gagal 3 kali, 	Prioritas : <u>4</u> Nama pengirim : <u>AZALEA</u> Nama Penerima : <u>RIFKI</u> Pesan: <u>Data Test</u> <u>1, karakter</u> <u>% dan @</u> <u>dapat</u> <u>dimasukkan</u> <u>sebagai</u> <u>pesan.</u>	Terbentuk di file pesan sebagai berikut: PRIORITAS XTITIKDUA 4 ENTER NAMA PENGIRIM XTITIKDUA AZALEA ENTER NAMA PENERIMA XTITIKDUA RIFKI ENTER PESAN XTITIKDUA DATA TEST 1 XKOMA KARAKTER XPERCENTAGE DAN XAT DAPAT DIMASUKKAN SEBAGAI PESAN#	Hasil yang keluar sesuai dengan hasil yang diharapkan.

			maka proses pengiriman pesan dibatalkan. Jika tidak pesan terkirim dan menampilkan informasi biaya pesan dan menyimpan sebagian data untuk statistik			
2	Hitung Harga	Menghitung harga dari pesan terakhir	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 1 untuk memilih modul Penulisan, lalu tekan ENTER 3) Ketik angka 2 untuk memilih Submenu Hitung Harga	[Menu di modul penulisan] Pilih menu: <u>2</u>	Harga: 14000 Jumlah Total Kata: 27 kata Kata Pendek: 16 Kata Panjang: 3 Tanda Baca: 4 Lain-lain: 4 Enter: 3	Hasil yang keluar sesuai dengan hasil yang diharapkan.
3	Statistik	Menampilkan statistik pesan	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 1 untuk memilih modul Penulisan, lalu tekan ENTER 3) Ketik angka 3 untuk memilih Submenu Statistik	[Menu di modul penulisan] Pilih menu: <u>3</u>	1. AZALEA 2 pesan 26 kata/pesan Rp 27300	Hasil yang keluar sesuai dengan hasil yang diharapkan.
4	Transmit semua pesan	Mentransmit pesan masing-masing 30 detik ke dalam folder inbox	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 2 untuk memilih modul Pengiriman, lalu tekan ENTER 3) Ketik angka 1	[Menu di modul pengiriman] Pilih menu: <u>1</u>	1 pesan telah terkirim	Hasil yang keluar sesuai dengan hasil yang diharapkan.

5	Statistik	Menampilkan statistik pesan	<p>untuk memilih Submenu Transmt Semua Pesan</p> <ol style="list-style-type: none"> 1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 2 untuk memilih modul Pengiriman, lalu tekan ENTER 3) Ketik angka 7 untuk memilih Submenu Statistik 	[Menu di modul pengiriman] Pilih menu: <u>7</u>	1. 2012-12-03 01 1 pesan	Hasil yang keluar sesuai dengan hasil yang diharapkan.
6	List Pesan	Menampilkan list pesan pada folder inbox	<ol style="list-style-type: none"> 1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 3 untuk memilih modul Penerimaan, lalu tekan ENTER 3) Ketik angka 1 untuk memilih Submenu List Pesan 	[Menu di penerimaan] Pilih menu: <u>1</u>	1. Inbox /121203053012 -AZALEA.txt	Hasil yang keluar sesuai dengan hasil yang diharapkan.
7.	Baca Pesan	Membaca pesan sesuai ID	<ol style="list-style-type: none"> 1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 3 untuk memilih modul Penerimaan, lalu tekan ENTER 3) Ketik angka 2 untuk memilih Submenu Baca Pesan 	Masukkan ID pesan: <u>1</u>	PRIORITAS: 1 Nama pengirim : AZALEA Nama Penerima : RIFKI Pesan: Data Test 1, karakter % dan @ dapat dimasukkan sebagai pesan.	Hasil yang keluar sesuai dengan hasil yang diharapkan.
8.	Hapus Pesan	Menghapus pesan sesuai ID	<ol style="list-style-type: none"> 1) Jalankan program, tekan ENTER untuk melewati informasi 	Masukkan ID pesan: <u>1</u>	Pesan [nama pesan] telah terhapus dari sistem.	Hasil yang keluar sesuai dengan hasil yang diharapkan.

			inisialisasi 2) Ketik angka 3 untuk memilih modus Penerimaan, lalu tekan ENTER 3) Ketik angka 3 untuk memilih Submenu Hapus Pesan			
9.	Statistik	Menampilkan statistik pesan	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 3 untuk memilih modus Penerimaan, lalu tekan ENTER 3) Ketik angka 7 untuk memilih Submenu Statistik Pesan	[Menu di modus penerimaan] Pilih menu: <u>7</u>	1. AZALEA 2/1 pesan	Hasil yang keluar sesuai dengan hasil yang diharapkan.
10.	Daftar Harga	Menampilkan list daftar harga	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modus Konfigurasi, lalu tekan ENTER 3) Ketik angka 1 untuk memilih Submenu Daftar Harga	[Menu di modus konfigurasi] Pilih menu: <u>1</u>	Jenis kata pendek seharga 500 Jenis kata panjang seharga 800 Jenis kata tanda baca seharga 200 Jenis kata lain-lain seharga 700 Jenis kata enter/baris baru seharga 0	Hasil yang keluar sesuai dengan hasil yang diharapkan.
11.	List Item Kamus	Menampilkan item tanda baca yang terdaftar pada kamus	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modus Konfigurasi, lalu tekan ENTER 3) Ketik angka 2 untuk memilih	[Menu di modus konfigurasi] Pilih menu: <u>2</u>	! - TANDASERU " - KUTIPGANDA dst..	Hasil yang keluar sesuai dengan hasil yang diharapkan.

			Submenu List Item Kamus			
12.	Penambahan Item Kamus	Menambahkan list item kamus	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modul Konfigurasi, lalu tekan ENTER 3) Ketik angka 3 untuk memilih Submenu Penambahan Item Kamus	Masukkan karakter khusus yang akan ditambahkan : { Masukkan kata untuk karakter khusus tersebut : <u>KURAWAL</u>	Berhasil ditambahkan ke kamus. Pada file eksternal yang menyimpan list kamus, terdapat tambahan karakter.	Hasil yang keluar sesuai dengan hasil yang diharapkan.
13.	Penghapusan Item Kamus	Menghapus list item kamus	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modul Konfigurasi, lalu tekan ENTER 3) Ketik angka 4 untuk memilih Submenu Penghapusan Item Kamus	Masukkan karakter khusus yang akan dihapus : {	Proses penghapusan berhasil. Pada file eksternal yang menyimpan list kamus, karakter yang diinput menjadi terhapus.	Hasil yang keluar sesuai dengan hasil yang diharapkan.
14.	List Pelanggan	Menampilkan list pelanggan yang terdaftar	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modul Konfigurasi, lalu tekan ENTER 3) Ketik angka 5 untuk memilih Submenu List Pelanggan	[Menu di modul konfigurasi] Pilih menu : <u>5</u>	1. AZALEA 2. RIFKI	Hasil yang keluar sesuai dengan hasil yang diharapkan.
15.	Penambahan Pelanggan	Menambahkan pelanggan	1) Jalankan program, tekan ENTER untuk melewati informasi	Masukkan nama pengguna baru : <u>DINAH</u>	Nama pengguna baru tersimpan dengan baik.	Hasil yang keluar sesuai dengan hasil yang diharapkan.

			inisialisasi 2) Ketik angka 4 untuk memilih modul Konfigurasi, lalu tekan ENTER 3) Ketik angka 6 untuk memilih Submenu Penambahan List Pelanggan			
16.	Penghapusan Pelanggan	Menghapus nama pelanggan dari daftar	1) Jalankan program, tekan ENTER untuk melewati informasi inisialisasi 2) Ketik angka 4 untuk memilih modul Konfigurasi, lalu tekan ENTER 3) Ketik angka 7 untuk memilih Submenu Penghapusan List Pelanggan	Masukkan nama yang akan dihapus: <u>DINAH</u>	Nama pengguna DINAH dihapus dari daftar.	Hasil yang keluar sesuai dengan hasil yang diharapkan

8 Pembagian Kerja dalam Kelompok

Nama	Bagian
Arief Rahman	ADT Priority Queue, ADT Pohon Biner, ADT List, ADT Mesin Karakter, ADT Mesin Kata, Encoder Decoder Telegram-Morse, Modus Pengiriman
Azalea Fisitania	Interface Master Program, File Eksternal
Dinah Kamilah	Modus Penulisan
Rifki Afina Putri	Modus Penerimaan, File Eksternal
Setyo Legowo	Modus Pengaturan, Penggabungan Master Program

*Catatan: Pembagian di atas adalah pembagian secara formal. Karena file master berupa shared file di Dropbox, tiap anggota bisa saling mengedit modul bila ada yang perlu diperbaiki

9 Lampiran

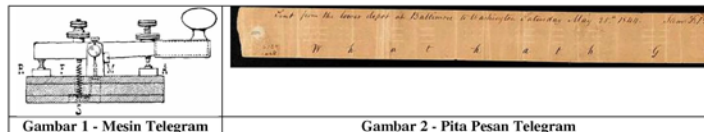
6.1 Deskripsi Tugas Besar

SISTEM TELEGRAM

Pengantar

Sebuah **sistem telegram** (*telegram system*) adalah sebuah sistem pengiriman pesan jarak jauh dimana pesan dikirimkan dengan signal listrik melalui sebuah kabel. Dalam mengirimkan pesan, sistem ini membutuhkan sebuah skema pengkodean yang dimengerti oleh pengirim dan penerima. Desain skema encoding pada sistem telegraph sangat bergantung pada kemampuan dan keterbatasan dari media pensignalan (i.e., kabel + signal listrik).

(Morse 1869) menyebutkan sistem telegram pertama ditemukan Schilling pada tahun 1832 saat dan pada tahun 1837 Morse telah berhasil melakukan percobaan mengirimkan dan merekam pesan telegram dengan sebuah mesin elektronik (**Gambar 1**). Pada mesin tersebut Morse menggunakan sebuah skema pengkodean, yang sekarang lazim disebut kode Morse, yang tercetak/terekam pada sebuah pita kertas (**Gambar 2**) yang selanjutnya akan diterjemahkan kembali dalam serangkainya alfabet.

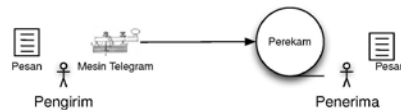


Akan tetapi kode morse hanya melakukan pengkodean terhadap simbol alfabet (i.e., a-z), sedangkan karakter lain (e.g., karakter khusus, tanda baca) haruslah direpresentasikan dalam alfabet sebelum dikirim melalui telegram. Berikut adalah sejumlah contoh representasi alfabet dari karakter lain-lain:

Karakter (non Alfabet)	Representasi Alfabet
1	sat
2	dua
!	tandas
@	at
...	...

Tabel 1 - Kamus Telegram

Sehingga secara umum (pada Gambar 3), sebuah pesan akan dibaca oleh operator pengirim dan dikodekan menjadi kode morse. Selanjutnya kode-kode tersebut dikirimkan melalui jaringan telegram ke penerima. Operator penerima akan menterjemahkan kode-kode morse tersebut sehingga menjadi sebuah pesan yang dapat langsung dibaca oleh penerima akhir dari pesan telegram.



Gambar 3 - Sistem Telegram

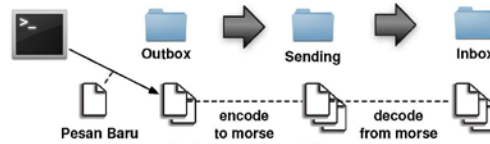
Pada umumnya, skema biaya dari sebuah telegram berdasar pada jumlah kata yang dikirimkan, dan pada umumnya kata-kata akan digolongkan menjadi sejumlah kategori harga berdasar pada panjang kata dan fungsi kata tersebut dalam pesan (e.g., tanda baca, karakter khusus, dll.), sebagai berikut:

Jenis Kata	Harga
Pendek (≤11 karakter)	500
Panjang (>11 karakter)	800
Tanda Baca (! , . ? / \ ` " : ;)	200
Lain-Lain (~ @ # \$ % ^ & * () > < - + =)	700
Enter/Baris Baru	0

Tabel 2 - Harga Telegram per Kata

Simulasi Sistem Telegram

Pada umumnya setiap pesan telegram, akan dikirimkan melalui sebuah jaringan kabel/telegram. Akan tetapi pada tugas ini anda diminta membangun simulasi sistem telegram dalam Bahasa C dengan menggantikan pengiriman/penerimaan pesan-pesan tersebut menjadi pemindahan sebuah file dari suatu folder ke folder lain berdasar status pengirimannya seperti pada Gambar 4.



Gambar 4 - Skema Simulasi Sistem Telegram

Pada simulasi ini, sistem pada Gambar 3 akan disimulasikan menjadi sebuah sistem piranti lunak/program yang memiliki tiga modul operasi (penulisan pesan, pengiriman pesan, dan penerimaan pesan). Secara umum pengirim akan menuliskan pesan telegram pada modul penulisan pesan, dan setiap pesannya akan tersimpan pada sebuah file di folder "Outbox". Pesan tersebut baru akan dikirim jika ada program yang berjalan pada modul pengiriman pesan, aktifitas ini disimulasikan dengan cara melakukan translasi pesan yang dituliskan pada modul pengiriman pesan kedalam kode morse dan diletakkan pada folder "Sending". Akhirnya pada modul penerimaan pesan, pesan yang ada pada folder "Sending" dibaca kembali menjadi pesan tekstual seperti yang terkirim pada awalnya.

Selain dari ketiga modul operasi, simulasi ini memiliki sebuah modul tambahan, modul pengaturan. Pada modul ini pengguna dapat mendefinisikan sejumlah parameter, seperti: kamus pemetaan, harga, dan data pelanggan. Pengguna dapat berganti modul penggunaan sesuai dengan kebutuhan/*use case* saat itu.

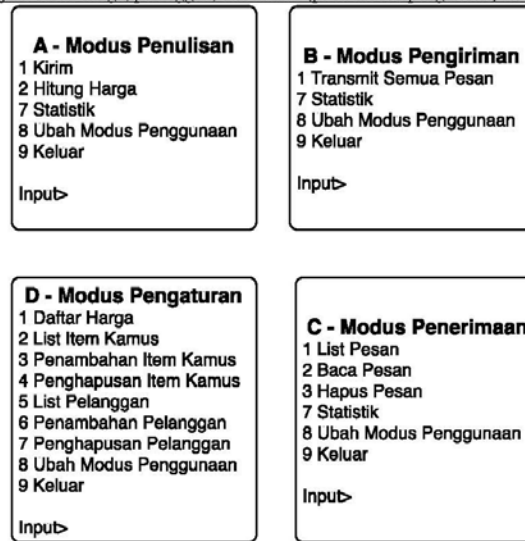
Interaksi Program dengan Pengguna

Ketika program dipanggil pertama kali, pengguna masuk ke *command prompt* yang ditandai dengan prompt ">" dan siap melakukan operasi simulasi dari sistem telegram pada modul pengiriman pesan. Perintah-perintah di lingkungan ini diketikkan setelah prompt "INPUT>". Jika dimasukkan pilihan "8" maka program akan memberikan sejumlah pilihan menu yang dapat mengubah program ke modul lain, seperti pada menu berikut:

```
A - Modul Penulisan
B - Modul Pengiriman
C - Modul Penerimaan
D - Modul Pengaturan

Masukkan modul yang dikehendaki:
```

Struktur menu lengkapnya dapat dilihat pada Gambar 5. Modus Pengiriman, Penerimaan, dan Pengaturan hanya dapat dijalankan jika daftar harga, pelanggan, dan kamus (pada modus pengaturan) telah terdefinisi seluruhnya.



Gambar 5 - Struktur Menu

Modus Penulisan

Modus ini berfungsi menerima masukan pesan dari operator, dan mengkonversikan ke dalam representasi yang dapat dikirimkan melalui telegram. Berikut adalah pilihan menu yang tersedia pada Modus Penulisan.

1. Kirim

Menu pilihan yang digunakan untuk mengirimkan pesan dari seorang pelanggan ke pelanggan lainnya. Pada menu ini operator telegram/pengguna diminta memasukkan: nama pengirim, nama penerima, dan pesan¹.

```

Prioritas      : -
Nama Pengirim  : Budi
Nama Penerima  : Iwan
Pesan          : Urgent, harap segera pulang. Ayah sakit keras.
  
```

Proses pengiriman ini akan sukses jika masukan pengguna telah benar.

- Nama pengirim dan penerima telah terdaftar pada sistem sebagai pelanggan
- Seluruh tanda baca dan karakter khusus telah terdefinisi di kamus

Jika hal-hal tersebut tidak terpenuhi proses pengiriman gagal dan sistem akan meminta operator untuk melakukan koreksi hingga kedua kriteria tersebut terpenuhi. Jika hingga 3 kali koreksi masukan tersebut masih belum valid maka sistem akan membatalkan pengiriman pesan ini, dan jika operator masih ingin mengirimkan pesan tersebut maka ia harus memulai dari awal.

Secara internal, menu ini akan menyimpan pesan tersebut pada folder "Outbox" dengan nama file: YYMMDDHHMMSS-<pengirim>.txt dimana <pengirim> adalah nama pengirim dan YY adalah tahun pengiriman, MM adalah bulan pengiriman, DD adalah tanggal pengiriman, HH adalah jam pengiriman, MM adalah menit pengiriman, dan SS adalah detik pengiriman. Pesan tersebut harus kedalam bentuk yang dapat dikirimkan via telegram/kode morse. Contoh hasil konversi pesan diatas dapat dilihat sebagai berikut:

¹ Pesan ditulis dengan case insensitive.

```
Nama Pengirim TitikDua Budi ENTER Nama Penerima TitikDua Iwan ENTER
Pesan TitikDua Urgent KOMA harap segera pulang TITIK Ayah sakit keras
TITIK
```

Setelah pesan sukses disimpan maka program mencetak ke layar jumlah kata dari masing-masing jenis kata (pada Tabel 2) dan biaya atas pesan yang baru dikirimkan oleh pengirim.

Catatan: Isian Prioritas hanya relevan jika anda berniat mengerjakan salah satu item bonus perihal *priority-queue*.

2. Hitung Harga

Mencetak informasi perintah pesan terakhir yang dikirim. Informasi itu meliputi: pengirim, penerima, perhitungan harga dan jumlah kata dari masing-masing jenis kata dari pesan terakhir yang dikirim.

```
Info harga dari pesan terakhir
Nama Pengirim: Budi
Nama Penerima: Iwan
Harga 8200
Jumlah Total Kata 22 kata
    Kata Pendek 14
    Kata Panjang 0
    Tanda Baca 6
    Lain-lain 0
Enter 2
```

7. Statistik

Mencetak statistik pesan yang ditulis oleh seluruh pelanggan yang pernah dikirim melalui sistem ini. Statistik pada modul ini meliputi: Nama pengirim, Jumlah Pesan yang dikirim, Rata-rata jumlah total kata dalam sebuah pesan, dan total harga dari pesan-pesan yang pernah dikirim.

```
Budi  20 Pesan    30 kata/pesan    160000
Iwan  2 Pesan    60 kata/pesan    40000
...
```

Catatan: Tidak terdapat batasan maksimum jumlah pelanggan pada statistik ini. Statistik ini harus disimpan secara persisten pada sebuah file.

8. Ubah Modus Penggunaan

Menu untuk berpindah modus penggunaan dari sistem simulasi telegram.

```
A - Modus Penulisan
B - Modus Pengiriman
C - Modus Penerimaan
D - Modus Pengaturan

Masukkan modus yang dikehendaki:
```

9. Keluar

Menu untuk keluar dari program sistem simulasi telegram.

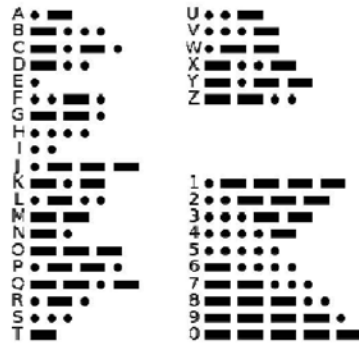
Modus Pengiriman

Modus penggunaan ini berfungsi untuk mengirimkan pesan yang telah terkonversikan ke pengguna. Berikut adalah pilihan menu yang tersedia pada modus ini.

1. Transmit Semua Pesan

Menu ini melakukan pengiriman seluruh pesan yang telah ditulis ke tujuan. Pada simulasi ini, pengiriman sebuah pesan melalui jaringan telegram disimulasikan sebagai tindakan mentranslasikan sebuah pesan kedalam

kode morse dan meletakkan pada folder *Sending* dengan nama file yang sama. Setelah proses tersebut sukses dilakukan, maka file pada folder *Outbox* harus dihapus.



Gambar 6 - Kode Morse

Saat menu ini dipilih, program akan memuat/me-load seluruh file-file pesan pada folder *Outbox* sebagai antrian pekerjaan yang harus dikirimkan. Antrian pekerjaan pada simulasi ini terurut berdasar waktu penulisan pesan (dapat dilihat dari nama file). Pada simulasi ini, media pengiriman hanya mampu mengirimkan **satu pesan pada tiap 30 detik**.

Catatan: Ketika menggunakan *priority-queue* (bonus), urutan antrian memperhatikan isian prioritas.² Jika terdapat sejumlah pesan yang memiliki prioritas yang sama, maka urutan akan didasarkan pada waktu penulisan pesan.

7. Statistik

Mencetak statistik jumlah pengiriman pesan pada tiap jamnya.

```
2012-10-28 12      20 Pesan
2012-10-28 13       2 Pesan
2012-10-29 14       4 Pesan
...
```

Catatan: rekaman ini akan disimpan secara persisten pada sebuah file.

8. Ubah Modus Penggunaan

Sama dengan menu pada modus pengiriman

9. Keluar

Sama dengan menu pada modus pengiriman

Modus Penerimaan

1. List Pesan

Menu ini menampilkan list pesan-pesan yang telah terkirim. Secara internal program simulasi melakukan konversi file dengan representasi kode morse pada folder "*Sending*" kedalam representasi tekstual (String alfabet) dan diletakkan pada folder "*Inbox*", dan jika proses ini sukses maka file pada folder "*Sending*" harus dihapus. Kemudian program akan mencetak ke layar nama file dari pesan-pesan tersebut.

```
1. 121028121300-Budi.txt
2. 121028121400-Iwan.txt
3. 121029121500-Budi.txt
...
```

² Simulasi ini menggunakan 4 kelas prioritas: Biasa, Segera, Penting, dan Kritisal.

2. Baca Pesan

Menu ini akan membutuhkan masukan id pesan dari pengguna. Jika id terdefinisi maka pesan akan dicetak ke layar, dengan merepresentasi pesan sebagaimana pesan tersebut dikirimkan. Dengan kata lain akan mengubah kembali representasi alfabet dari tanda baca, karakter khusus, dan enter menjadi simbol-simbol semual, sehingga lebih mudah dibaca oleh pengguna.

Contoh:

```
Input> 2
Masukkan Id Pesan> 1
```

sehingga pesan³:

```
Nama Pengirim TitikDua Budi ENTER Nama Penerima TitikDua Iwan ENTER
Pesan TitikDua Urgent KOMA harap segera pulang TITIK Ayah sakit keras
TITIK
```

akan ditampilkan menjadi:

```
Nama Pengirim : Budi
Nama Penerima : Iwan
Pesan : Urgent , harap segera pulang . Ayah sakit keras .
```

3. Hapus Pesan

Menu ini akan membutuhkan masukan id pesan dari pengguna. Jika id terdefinisi maka pesan akan dihapus dari folder "Inbox" dan tidak akan muncul lagi pada list pesan.

Contoh:

```
Input> 3
Masukkan Id Pesan> 1
```

Pesan 121028121300-Budi.txt telah terhapus dari sistem.

7. Statistik

Mencetak statistik pembacaan pesan yang telah diterima. Statistik pada modul ini meliputi: Nama Penerima, Jumlah Pesan yang diterima dan Jumlah pesan yang dibaca.

```
Iwan 10/20 Pesan
Budi 2/2 Pesan
...
```

Catatan: Rekaman ini disimpan secara persisten pada sebuah file..

8. Ubah Modus Penggunaan

Sama dengan menu pada modus pengiriman.

9. Keluar

Sama dengan menu pada modus pengiriman.

Modus Pengaturan

Seluruh data pada modul ini harus tersimpan secara persisten di file.

1. Daftar Harga

Menampilkan list harga dari jenis-jenis kata.

Catatan: Konfigurasi pada Tabel 2 harus *configurable* dengan mengubah harga dari sebuah jenis kata pada file konfigurasi.

³ Perbedaan case (huruf besar/kecil) pada pesan di dokumen ini hanya digunakan untuk memudahkan pembacaan. Pesan pada file di outbox atau inbox disimpan secara case insensitive (huruf besar/kecil semua).

2. List Item Kamus

Mencetak ke layar daftar item yang ada di kamus, sebagai contoh sebagai berikut:

```
' - apostroph
~ - tilde
! - tandaseru
@ - at
# - hash
$ - dollar
% - percentage
^ - caret
& - and
* - asterisk
( - kurungbuka
) - kurungtutup
" - kutipganda
```

List item kamus dapat diubah berdasar

3. Penambahan Item Kamus

Menambah item baru ke dalam kamus, jika item tersebut belum terdefinisi ke dalam kamus.

4. Penghapusan Item Kamus

Menghapus item yang telah terdefinisi pada kamus.

5. List Pelanggan

Mencetak ke layar list pelanggan yang telah terdaftar di sistem.

6. Penambahan Pelanggan

Mendaftarkan pelanggan baru ke dalam sistem, jika pelanggan belum terdaftar dalam sistem.

7. Penghapusan Pelanggan

Menghapus entri pelanggan yang terdefinisi dalam sistem.

8. Ubah Modus Penggunaan

Sama dengan menu pada modus pengiriman.

9. Keluar

Sama dengan menu pada modus pengiriman.

Deskripsi Pengerjaan Tugas

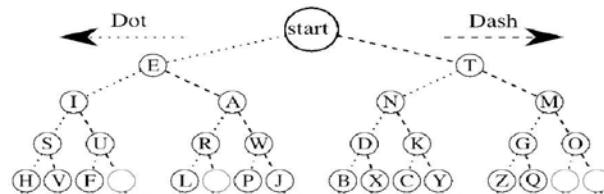
- Beberapa batasan yang berlaku untuk tugas ini (sebelum pengerjaan bonus):
 - Semua proses akuisi input dari pengguna harus dilakukan dengan mesin kata.
 - Minimum tanda baca dan simbol yang harus di dukung: '~ ! @ # \$ % ^ & * () \ / ? > < . , ; : " - + =
- Tugas ini menggunakan banyak ADT dan mesin yang pernah diajarkan sampai saat tugas ini diberikan dengan modifikasi yang diperlukan.
- Tugas harus diserahkan per kelompok (maksimum 5 orang dalam satu kelas), dengan cara pengelompokan yang akan ditentukan. Setiap mahasiswa harus memahami spesifikasi, dan akan diberikan 1 sesi khusus untuk klarifikasi spesifikasi.
- Waktu dan tatacara penyerahan serta pemeriksaan tugas akan diumumkan di situs <http://kuliah.itb.ac.id> pada *link* kuliah IF2030/Algoritma dan Struktur Data.
- Pembagian tugas (penanggung jawab masing-masing modul program) harus dinyatakan dengan jelas dalam penyerahan tugas.
- Deadline: Senin, 3 Desember 2012 17.00 WIB

Deliverables

1. *Source code, well commented*, dengan pembagian *file* yang sesuai standar ([namaADT].h, [namaADT].c, [driver].c). Primitif dalam ADT hanya primitif yang dipakai dalam program. Satu modul ADT disimpan dalam satu folder yang terpisah.
 2. Laporan yang berisi antara lain:
 - Detil spesifikasi.
 - Sketsa struktur data internal.
 - Data test.
 - *Test script*, yaitu urutan tes yang “masuk akal” dan mencakup minimal semua fasilitas yang harus dites serta sesuai spesifikasi di atas.
 - Notulen setiap pertemuan untuk klarifikasi spesifikasi yang dilakukan per kelompok.
- Akan diberikan *template* laporan.

Bonus

- Pengkodean dan peng-dekodean kode morse direpresentasikan secara *binary-search-tree*.



- Gambar 7 - Pengkodean Kode Morse dalam bentuk Pohon

- Antrian dari pengiriman pesan adalah merupakan dapat diubah menjadi *priority-queue* hanya dengan mengganti library antrian/queue pada saat proses objek linking.
- Simulasi ini dapat berjalan dengan menjalankan 3 (tiga) *instance* dari program ini yang bekerja secara paralel dalam sebuah host yang sama, asalkan ketiga program tersebut berjalan dengan modus yang berbeda-beda: penulisan, pengiriman, dan penerimaan.
- Validasi untuk input perintah (*command*) maupun file.
- Sorting dari data statistik berdasarkan field tertentu.
- Penanganan khusus terhadap kata pesan yang sama dengan kata dalam kamus, misalnya: TITIK (bisa merupakan nama orang, bisa juga merupakan tanda baca yang ada di kamus).

6.2 Notulen Rapat

Rapat ke-1

Tempat : Sekre 2 HMIF

Hari, Tanggal : Senin, 5 November 2012

Waktu : 20.00 – 21.00

Anggota Kelompok yang hadir : Arief, Azalea, Dinah, Rifki, Setyo

Isi rapat : - Pembahasan spesifikasi tugas, perencanaan cara kerja program yang akan dibuat.
- Pembagian tugas kepada masing-masing anggota kelompok oleh ketua (Setyo L.)

6.3 Log Activity Anggota Kelompok

Nama	Waktu	Kegiatan
Arief Rahman	16/11/2012	<ul style="list-style-type: none">- Memasukkan ADT Priority Queue (Tested)- Mengubah infotype menjadi string untuk disesuaikan dengan spesifikasi- Memasukkan ADT List Rekursif (Tested)- Memasukkan ADT List Rekursif- ADT belum lengkap, lihat bagian binary search tree- ADT sudah dapat membentuk tree dari pitakar (kode MORSE)- ADT belum dites secara keseluruhan
	22/11/2012	<ul style="list-style-type: none">- Konflik makro dan tipe di list dan queue telah diperbaiki

Azalea Fisitania	12:28 WIB	
	26/11/2012	- ADT untuk proses encode dan decode telah selesai dibuat
	19:00 WIB	- Sedikit perubahan di ADT Pohon Biner
	28/11/2012	- Perbaikan pada pitakar.txt (berisi kode Morse)
	19:10 WIB	
	29/11/2012	- Perbaikan terhadap ADT Pohon Biner
	12:53 WIB	- ADT Pohon Biner telah dites secara keseluruhan
	01/12/2012	- Transmit pesan dapat digunakan kembali
	18:00 WIB	- [22:00] Transmit 30 detik pesan sudah dapat digunakan
	02/12/2012	- Penjelasanodus pengiriman telah ditambahkan
Dinah Kamilah Ulfa	22.34 WIB	- Kurung siku [], Kurung kurawal { }, Dash _, dan Vertical Bar telah dimasukkan ke dalam inisialisasi kamus
		- Inisialisasi kamus telah diurutkan berdasarkan kode ASCII
		- Penjelasan inisialisasi konfigurasi telah ditambahkan
		- Penjelasanodus konfigurasi telah ditambahkan
		- Semua file header telah diberi penjelasan
		- Pemberian nama file masih ada masalah (testing di Windows)
	12/11/2012	- Mulai membuat Encoder Morse melalui Binary Search Tree untuk Modus Pengiriman
	18/11/2012	- Mulai membuat ADT Priority Queue
	22/11/2012	- Peralihan kerja ke interface. Interface umum selesai, tinggal difollow up bila ada perubahan. Debug hirearki menu secara umum yang akan ditampilkan pada user
	12:28 WIB	
Rifki Afina Putri	01/12/2012	- Menulis laporan
	18.00 WIB	
	02/12/2012	- Tampilan di program sudah dirapikan
	06:25 WIB	- (Test) Pesan belum dapat menyertakan angka karena bingung membedakan satuan dengan puluhan
		- File eksternal yang pesan Azal formatnya .txtWO
		- Rekap log act anggota, menulis laporan tubes
	03/12/2012	- Format file eksternal sudah menjadi .txt
	14:20 WIB	- Rekap final log act anggota dan dokumen asistensi
	26/11/2012	- Modus Penulisan bertambah beberapa fungsi
	19:00 WIB	
	30/11/2012	- Modus Penulisan bagian kirim dan baca pesan terakhir sudah bisa digunakan
	03:30 WIB	- Statistik Modus penulisan belum bisa digunakan
		- Statistik Penulisan sudah dapat digunakan
	01/12/2012	
	18:00 WIB	
	19/11/2012	- Mulai membuat modus penerimaan
	15:30 WIB	
	26/11/2012	- Modus Penerimaan selesai namun belum digabung dengan program utama (kurang pada bagian statistik)
	16:30 WIB	

Setyo Legowo	30/11/2012 16:30 WIB	- Modus Penerimaan selesai dan telah digabungkan dengan program utama. Tetapi bagian statistik belum dapat digunakan.
	01/12/2012 17:00 WIB	- Penyelesaian bagian statistik pesan, awalnya program tidak bisa berjalan ketika mengupdate statistik saat pembacaan pesan. Setelah beberapa perbaikan, statistik dapat berjalan dan modus penerimaan dapat digunakan. - [20.00] Proses membaca pesan masih error, yaitu id pesan pada List pesan tidak sesuai dengan pesan yang keluar di bagian baca pesan - Proses membaca pesan masih error, yaitu id pesan pada List pesan tidak sesuai dengan pesan yang keluar di bagian baca pesan
	02/12/2012	- Menuliskan penjelasan ADT yang digunakan serta algoritma menarik pada laporan
	03/12/2012	- Menuliskan data test dan test script
	30/11/2012 16:30 WIB	- Modus Penerimaan selesai dan telah digabungkan dengan program utama. Tetapi bagian statistik belum dapat digunakan.
	1/12/2012 17:00 WIB	- Penyelesaian bagian statistik pesan, awalnya program tidak bisa berjalan ketika mengupdate statistik saat pembacaan pesan. Setelah beberapa perbaikan, statistik dapat berjalan dan modus penerimaan dapat digunakan.
	26/11/2012 14:29 WIB	- MARK diganti dengan karakter pagar
	26/11/2012 18:00 WIB	- BACKUP source code pertama dan kedua
	27/11/2012 00.53 WIB	- Saat file digabung seluruhnya dan berhasil dicompile, saat dijalankan tidak mengeluarkan output apapun seperti perulangan tanpa henti.
	27/11/2012 10:10 WIB	- Modus konfigurasi selesai dan dapat digunakan.
	30/11/2012 03:30 WIB	- Modus Penulisan bagian kirim dan baca pesan terakhir sudah bisa digunakan - Statistik Modus penulisan belum bisa digunakan
	30/11/2012 03:30 WIB	- Folder compile menjadi 1, yaitu folder "bin". Maka didalamnya terdapat "main_program" yang dapat dijalankan di OS Windows dan OS Linux
	01/12/2012 07:34 WIB	- Modus Pengiriman untuk transmit pesan selesai, tinggal yang mengirim per 30 detik. - [10:33 WIB] Transmit pesan tidak dapat dipakai lagi - [18:00] Transmit pesan dapat digunakan kembali - Statistik Penulisan sudah dapat digunakan - [22:00 WIB] Statistik pengiriman dan penerimaan sudah dapat digunakan

	<ul style="list-style-type: none"> - Proses membaca sudah dapat digunakan dan sudah disesuaikan di List pesan, baca pesan, dan hapus pesan - Program sudah dapat digunakan sepenuhnya, hanya tinggal merapihkan saja. Beberapa bug mungkin masih terjadi
02/12/2012 11:32 WIB	<ul style="list-style-type: none"> - Transmit pesan 30 detik sudah diperbaiki - Tampilan di program sudah dirapikan - Transmit 30 detik pesan sudah dapat digunakan
03/12/2012 00:04 WIB	<ul style="list-style-type: none"> - Perbaiki string penamaan nama file untuk proses pengiriman - [02:00 WIB] Segmentation fault atau closing program saat setelah mengirim pesan dan setelah membaca list daftar karakter - [02:21] Closing program mungkin terjadi saat update statistik pelanggan dan list item kamus - [04:05 WIB] Program fix berjalan dengan baik - Bug terdapat pada List Item. Jumlah karakter yang terdaftar pada file lebih dari besar tabel kontingunya

6.4 Asistensi

Asisten: Rendy Bambang Junior / 13509036

Asistensi 1 20 November 2012 12.00 WIB Selasar Basis Data	<ol style="list-style-type: none"> 1. ID Pesan? Apakah diminta masukan dari user? → Berupa nomor urutan pesan yang bisa dilihat di List Pesan 2. Menampilkan list pesan? → Boleh memakai file eksternal, tapi sebaiknya dimasukkan ke dalam suatu variabel, disimpan pada array 3. File eksternal yang ada di folder-folder? (memastikan persepsi) → Pesan manusia (masukan user) -> Telegram (simpan di Outbox) -> Morse (simpan di Sending) -> Telegram (simpan di Inbox) -> Pesan manusia (keluaran program) → Berarti butuh encode-decode telegram dan encode-decode morse 4. Cara menjalankan 3 command sekaligus? → Sulit, mungkin memakai inter-processing communication 5. Membedakan "Titik" nama orang dengan TITIK karakter "." → Buat simbol sendiri untuk membedakan
Asistensi 2 30 November 2012 15.00 WIB Lab Programming	<ol style="list-style-type: none"> 1. Cara setting agar mengirim tiap 30 detik? → Pengulangan mengecek sistem waktu, coba googling 2. Linux atau Windows? → Tidak boleh di Windows. Sebenarnya boleh, tapi akan ada pengurangan nilai 3. Statistik pesan? → Direkam di file eksternal