

**Tugas Besar III IF2211 Strategi Algoritma
Semester II Tahun 2020/2021**

Penerapan String Matching dan Regular Expression
dalam Pembangunan Deadline Reminder Assistant



Waktu Mulai:

Rabu, 7 April 2021, 19.50 WIB

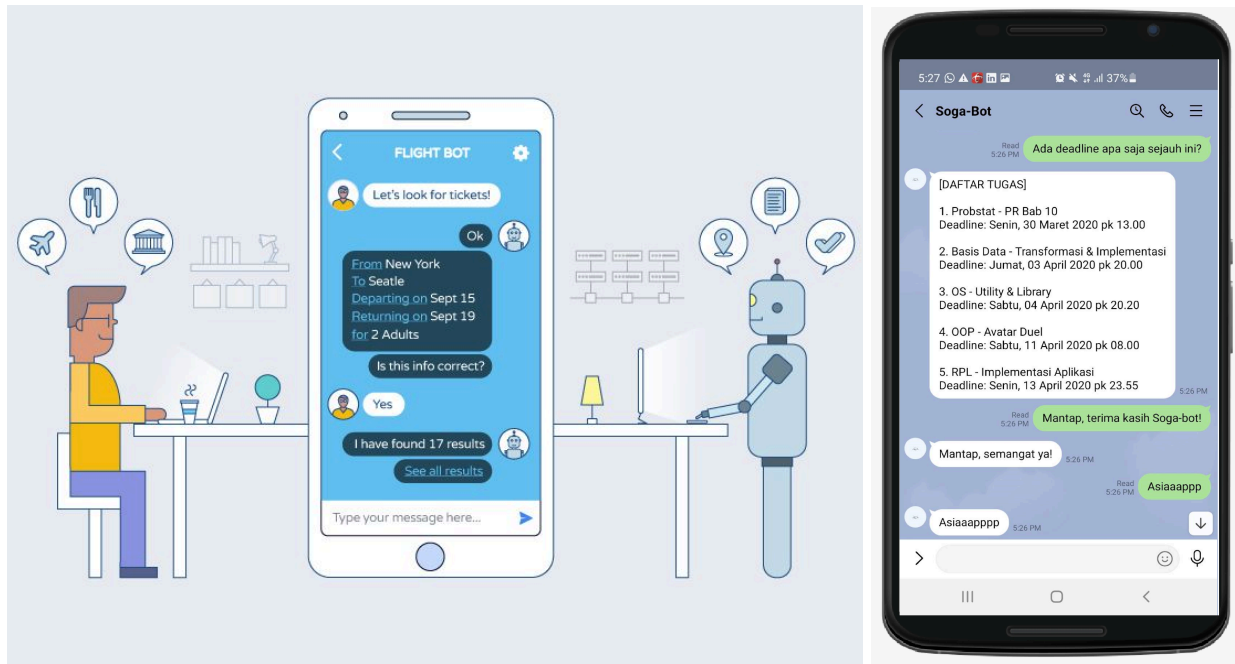
Waktu Akhir:

Rabu, 28 April 2021, 23.59 WIB

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Latar Belakang

Semakin hari tugas-tugas di Teknik Informatika Semester 4 semakin bertambah banyak, hal ini tentu menambah kegiatan dan pekerjaan yang harus dilakukan mahasiswa. Tidak jarang ada tugas yang terlupakan karena mahasiswa sulit untuk mengingat semua tugas dan *deadline* yang ada. Oleh karena itu, mahasiswa Teknik Informatika berniat untuk membuat suatu *Google Assistant* sederhana berupa Deadline Reminder Assistant atau Asisten Peningkat Deadline.



Gambar 1. Ilustrasi chatbot dan Asisten Peningkat Deadline

Sumber : <https://id.pinterest.com/pin/824299538024636729/> dan dokumentasi pribadi

Kita seringkali mendengar teknologi atau aplikasi seperti chatbot, LINE Bot, atau Google Assistant, ketiganya merupakan agen cerdas yang meniru kemampuan manusia untuk melakukan percakapan dengan *user*. Kehadiran chatbot dapat membantu kehidupan manusia, khususnya dalam penyajian informasi yang diperlukan *user* dan menjawab berbagai pertanyaan yang ditanyakan oleh *user*. Secara spesifik dalam konteks Asisten Peningkat Deadline ini, chatbot akan menjawab pertanyaan mahasiswa yang sering ditanyakan seperti *deadline* seminggu ke depan, *deadline* pada bulan ini, dan task penting lainnya yang perlu dilakukan. Chatbot ini akan membantu *user* agar tidak melewatkan *deadline* tugas lagi.

Deskripsi Tugas

Dalam tugas besar ini, Anda akan diminta untuk membangun sebuah *chatbot* sederhana yang berfungsi untuk membantu mengingat berbagai deadline, tanggal penting, dan task tertentu kepada pengguna. Dengan memanfaatkan algoritma String Matching dan Regular Expression, Anda dapat membangun sebuah *chatbot* interaktif sederhana layaknya Google Assistant yang akan menjawab segala pertanyaan Anda terkait informasi deadline tugas-tugas yang ada.

Fitur-Fitur Aplikasi

Deadline Reminder Assistant akan dibangun dengan sistem **Question and Answer** dimana pengembang diharapkan sudah menyediakan kumpulan formula tertentu untuk melakukan pendeteksian setiap perbedaan command atau perintah pada aplikasi chatbot. Berikut merupakan runtutan fitur yang dimiliki oleh Deadline Reminder Assistant tersebut:

1. Menambah *task* baru

Pada fitur ini, suatu kalimat dapat diklasifikasikan sebagai suatu *task* apabila mengandung **semua komponen** berikut ini:

- Tanggal (format dibebaskan)
- Kode Mata Kuliah / Nama Mata Kuliah (dibebaskan)
- Jenis Tugas (berdasarkan daftar **kata penting** yang sudah disediakan)
- Topik Tugas (tidak ada batasan)

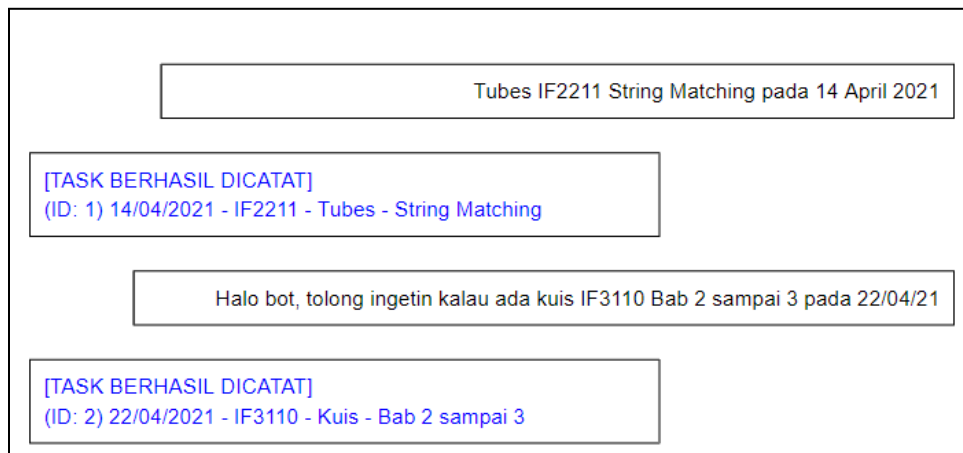
Poin a sampai d dibuat menggunakan **regular expression** sehingga masukan kalimat mirip dengan kalimat yang digunakan sehari - hari

Jika pesan berhasil dikenali oleh assistant, maka assistant akan mengirim pesan balasan yang berisi ID (sesuai urutan task diinput), tanggal, kode mata kuliah, jenis tugas, dan topik tugas. Contoh pesan balasan dari bot sebagai berikut

[TASK BERHASIL DICATAT]

(ID: 1) 14/04/2021 - IF2211 - Tubes - String matching

Berikut merupakan contoh interaksi pada *chatbot*



2. Melihat daftar *task*

Dalam fitur ini, *chatbot* diharapkan dapat menampilkan *task* sesuai dengan kriteria yang diberikan pengguna. Kriteria yang dapat diberikan oleh pengguna adalah:

- a. Menampilkan seluruh *task* yang sudah tercatat oleh assistant
Contoh perintah yang dapat digunakan: "Apa saja deadline yang dimiliki sejauh ini?"
- b. Menampilkan *task* berdasarkan periode waktu dengan rincian:
 - i. Pada periode tertentu (DATE_1 until DATE_2)
Contoh perintah yang dapat digunakan: "Apa saja deadline antara DATE_1 sampai DATE_2?"
 - ii. N minggu ke depan
Contoh perintah yang dapat digunakan: "Deadline N minggu ke depan apa saja?"
 - iii. N hari ke depan
Contoh perintah yang dapat digunakan: "Deadline N hari ke depan apa saja?"
 - iv. Hari ini
Contoh perintah yang dapat digunakan: "Apa saja deadline hari ini?"
- c. Menampilkan *task* berdasarkan jenisnya (kata penting) dengan kriteria:
 - i. *Task* merupakan kata yang sudah didefinisikan sebelumnya
 - ii. *User* dapat melihat daftar *task* dari suatu jenis *task* tertentu
Contoh: "3 minggu ke depan ada kuis apa saja?", maka chatbot akan menampilkan daftar kuis selama 3 minggu kedepan

Selama proses penggunaan *chatbot*, eksekusi perintah pengguna bisa mencakup ketiga poin sekaligus sehingga formula pengenalan *command* sebaiknya dibuat sebagai satu kesatuan utuh. Berikut merupakan contoh interaksinya

Apa saja deadline yang dimiliki sejauh ini?

[Daftar Deadline]

- (ID: 1) 14/04/2021 - IF2211 - Tubes - String Matching
- (ID: 2) 22/04/2021 - IF3110 - Kuis - Bab 2 sampai 3

Apa saja deadline antara 03/04/2021 sampai 15/04/2021?

[Daftar Deadline]

- (ID: 2) 14/04/2021 - IF3110 - Kuis - Bab 2 sampai 3

Deadline 3 minggu ke depan apa saja?

[Daftar Deadline]

- (ID: 1) 14/04/2021 - IF2211 - Tubes - String Matching
- (ID: 2) 22/04/2021 - IF3110 - Kuis - Bab 2 sampai 3

Deadline 1 hari ke depan apa saja?

Tidak ada

3 minggu ke depan ada kuis apa saja?

[Daftar Deadline]

- (ID: 2) 22/04/2021 - IF3110 - Kuis - Bab 2 sampai 3

Selain itu, hal lain yang perlu diperhatikan adalah perintah yang digunakan pengguna tidak harus selalu sama, asal perintah mengandung kata kunci yang ditentukan (kata kunci tiap perintah bisa ditentukan sendiri). Sebagai contoh kedua kalimat di bawah ini memberikan output yang sama

“Apa saja **deadline** antara **03/04/2021** sampai **15/04/2021**?”

“Antara **03/04/2021** dan **15/04/2021** ada **deadline** apa saja ya?”

3. Menampilkan *deadline* suatu *task*

Pada fitur ini, *deadline* yang dapat ditampilkan hanya berlaku untuk *task* yang bersifat tugas atau *task* yang memiliki *deadline*. Berikut merupakan contoh interaksi pada *chatbot*

The screenshot shows a chatbot interface with a white background and a thin black border. Inside, there are two rectangular boxes. The top box is white with a thin black border and contains the text "Deadline tugas IF2211 itu kapan?". The bottom box is also white with a thin black border and contains the date "14/04/2021" in blue text.

4. Memperbarui task

Fitur ini digunakan untuk memperbarui *deadline* dari suatu *task* seperti bila ada kasus pengunduran *deadline*. Perintah yang dapat diberikan pengguna meliputi 1 kata kunci untuk memperbarui *task* dan nomor *task* tertentu. Contoh perintah yang dapat digunakan pengguna adalah

“Deadline task X diundur menjadi 28/04/2021” dimana X merupakan nomor ID suatu *task*

selanjutnya *chatbot* dapat menampilkan pesan berhasil apabila *task* berhasil diperbarui dan pesan eror apabila *task* gagal diperbarui.

5. Menandai *task* yang sudah selesai dikerjakan

Fitur ini dibuat untuk menandai *task* yang sudah selesai dikerjakan untuk dihilangkan agar tidak ditampilkan lagi pada daftar *task*. Contoh perintah yang dapat digunakan adalah

“Saya sudah selesai mengerjakan task X” dimana X merupakan nomor ID dari suatu *task*

chatbot dapat menampilkan pesan berhasil apabila *task* berhasil diperbarui dan pesan eror apabila *task* gagal diperbarui.

6. Menampilkan opsi *help*

Fitur ini berfungsi untuk menampilkan perintah apa saja yang bisa diberikan oleh pengguna. *Chatbot* akan menampilkan hasil berupa daftar kata/kalimat yang bisa digunakan oleh pengguna untuk memberikan perintah kepada *chatbot*. Berikut merupakan contoh dari fitur ini

Apa yang bisa assistant lakukan?

[Fitur]

1. Menambahkan task baru
2. Melihat daftar task
3.
4. (dan seterusnya)

[Daftar kata penting]

1. Kuis
2. Ujian
3. Tupil
4. Tubes
5.

7. Mendefinisikan list kata penting

Fitur ini dibuat agar *chatbot* dapat mengenali kata penting ketika pengguna memberikan suatu perintah. *Chatbot* harus memiliki minimal 5 kata penting yang berbeda, seperti “Kuis”, “Ujian”, “Tupil”, “Tubes”, “Praktikum”, dan lain - lain. Kata penting akan digunakan untuk menentukan jenis dari *task*, daftar kata penting tidak perlu dibuat dinamis, cukup dibuat statik saja.

8. Menampilkan pesan error.

Fitur ini akan aktif ketika pengguna memberikan sebuah perintah yang tidak dikenali oleh *chatbot*. Perintah yang tidak dikenali merupakan perintah diluar apa yang sudah dijelaskan pada poin 1 - 6. Pesan error dibebaskan sesuai kreativitas mahasiswa. Berikut merupakan contoh interaksi yang mengandung pesan error

Apakah mayones sebuah instrumen?

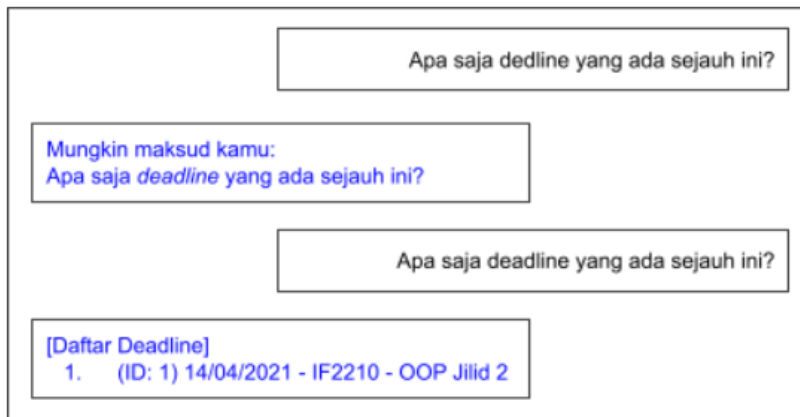
Maaf, pesan tidak dikenali

Halo, apa kabar?

Maaf, pesan tidak dikenali

9. **(Bonus)** *Chatbot* dapat memberikan rekomendasi kata

Pada fitur bonus ini, *chatbot* diharapkan dapat memberikan rekomendasi kata apabila pengguna *typo* saat menuliskan perintah. Rekomendasi dapat diberikan jika perintah dari pengguna *mismatch* dengan daftar kata tetapi masih memiliki tingkat kemiripan diatas 75%. Berikut merupakan contoh interaksinya



Untuk membantu pengerjaan fitur ini, ada berbagai matriks yang dapat dimanfaatkan untuk mencari kemiripan kata, salah satunya adalah Levenshtein distance yang diukur melalui pendekatan *dynamic programming*. Anda dapat mempelajari Levenshtein distance melalui pranala [ini](#).

Spesifikasi Program

1. Aplikasi yang dibuat berbasis *command line*
2. Aplikasi harus menggunakan algoritma pencocokan string KMP, Boyer-Moore, dan Regex dengan menggunakan bahasa yang menunjang regular expression misal: Java, Python.
3. Penyimpanan data-data dan pengetahuan yang diperlukan oleh chatbot bisa didefinisikan melalui 2 cara, yaitu:
 - a. Membuat suatu database sederhana dengan menerapkan prinsip basis data. Implementasi skema database dibebaskan dan skema basis data tidak perlu dinormalisasi.
 - b. Menyimpannya dalam bentuk struktur data sendiri, pengambilan data dilakukan dengan menggunakan mekanisme load / save dari suatu file .txt. Struktur penyimpanan data dibebaskan.
4. Data yang diperlukan dan akan disimpan pada *chatbot* adalah sebagai berikut:
 - a. List kata-kata penting
 - b. Daftar task yang tercatat oleh chatbot
 - c. Data pendukung lainnya (kreativitas kelompok)
5. Pencocokan string dapat anda implementasikan sesuai kriteria berikut.
 - a. Deteksi perintah (contoh: "Apa saja deadline yang ada sejauh ini?") **tidak dilakukan secara exact matching** (input dibebaskan ke **user --bukan programmer--** selama

- mengandung kata kunci tertentu), anda dapat memanfaatkan regular expression dan string matching untuk mencari kata kunci dan melakukan pencocokan.
- b. Rekomendasi kata: pencocokan exact matching (KMP, Boyer-Moore) dimanfaatkan untuk menentukan tingkat kemiripan suatu kata di perintah. Anda dapat mengembangkan algoritma yang telah diajarkan untuk menentukan kemiripan string.
 - c. Pengekstrakan nilai-nilai berjenis numerik dan tanggal dilakukan dengan memanfaatkan Regular Expression.
6. **(Bonus)** Aplikasi dibuat dengan basis web

Catatan

1. Anda dapat menambahkan fitur lain yang menunjang pembuatan program (unsur kreativitas).
2. Tugas dikerjakan berkelompok, minimal 2 orang dan maksimal 3 orang, boleh lintas kelas namun **tidak boleh sekelompok dengan orang yang sama dengan tubes stima sebelumnya.**
3. Semua kelompok harap mengisi data kelompok mereka pada link <http://tiny.cc/PendataanStima3>
4. Anda harus membuat aplikasi dan program ini sendiri kecuali library file dan regex, tetapi belajar dari contoh-contoh program serupa yang sudah ada tidak dilarang (tidak boleh melakukan plagiasi source code dari program orang lain). Program harus dibuat oleh kelompok sendiri, tidak boleh sama dengan kelompok lain ataupun ambil dari internet.
5. Program harus modular dan mengandung komentar yang jelas.
6. Beri nama tokoh di chatbot ini, misalnya Sogabot, Hansbot, Awoobot, dll. Lengkapi *chatbot* dengan gambar avatarnya. Avatar yang dinamis (bibir bisa bergerak, gerakan tubuh) adalah kreativitas yang dihargai.
7. Batas akhir pengumpulan tugas adalah **Rabu, 28 April 2021 pukul 23.59 WIB.** Keterlambatan akan diberi penalti pengurangan skor yang cukup signifikan.
8. Semua pertanyaan menyangkut tugas ini dapat dikomunikasikan lewat QnA yang bisa diakses pada <http://tiny.cc/QnATubes3>
9. **Bonus (maksimal 15 poin):**
 - a. Mendeploy aplikasi web yang telah dibangun (hosting provider dibebaskan). Deployment website harus dipertahankan sampai demo tugas besar.

- b. Setiap kelompok membuat video aplikasi yang mereka buat kemudian mengunggahnya ke Youtube. Video yang dibuat harus memiliki audio dan menampilkan wajah dari setiap anggota kelompok. Pada waktu demo aplikasi di depan asisten, mahasiswa mengakses video Youtube tersebut dan memutarnya di depan asisten. Beberapa contoh video tubes tahun-tahun sebelumnya dapat dilihat di YouTube dengan menggunakan kata kunci “Tubes Stima”, “Tugas besar stima”, “strategi algoritma”, dll.
 - c. Membuat fitur rekomendasi kata apabila terdapat kesalahan kata pada perintah.
10. Pelaksanaan demo akan diinformasikan lebih lanjut setelah waktu pengerjaan tugas berakhir
11. Setiap anggota kelompok harus memahami seluruh program, termasuk bagian yang bukan bagian mereka.
12. Program disimpan dalam folder **Tubes3_NIM** dengan menggunakan anggota dengan NIM terkecil. Berikut merupakan struktur dari isi folder tersebut.
 - a. Folder **src** berisi **source code**
 - b. Folder **doc** berisi **laporan tugas besar** dengan format **nama_kelompok.pdf**
 - c. Folder **test** berisi **data-data** atau **pengetahuan** awal yang dimiliki oleh chatbot. Apabila menggunakan database lokal, hasil dump dapat disimpan disini.
 - d. README selengkap mungkin. Referensi README dapat diakses pada <https://github.com/ritaly/README-cheatsheet> atau referensi lain yang serupa.
13. Folder tersebut **di-zip** dengan format yang sama dengan nama folder. Link pengumpulan akan diberitahukan lebih lanjut oleh asisten.

Format Laporan

Cover: Cover laporan adalah foto anggota kelompok (foto berdua/bertiga). Foto ini menggantikan logo “gajah” ganesha.

Bab 1: Deskripsi tugas (dapat menyalin spesifikasi tugas ini).

Bab 2: Landasan Teori.

Deskripsi singkat algoritma KMP, BM, dan Regex

Penjelasan singkat mengenai chatbot

Bab 3: Analisis Pemecahan Masalah.

Langkah penyelesaian masalah setiap fitur

Fitur fungsional dan arsitektur chatbot yang dibangun

Bab 4: Implementasi dan pengujian.

Spesifikasi teknis program (struktur data, fungsi, prosedur yang dibangun)

Penjelasan tata cara penggunaan program (interface program, fitur-fitur yang disediakan program, dan sebagainya)

Hasil pengujian (*screenshot* antarmuka dan skenario yang memperlihatkan berbagai kasus yang mencakup seluruh fitur dalam chatbot)

Analisis hasil pengujian

Bab 5: Kesimpulan, saran, dan komentar/refleksi tentang tugas besar 3 ini.

Daftar Pustaka.

Keterangan

1. Laporan ditulis dengan bahasa Indonesia yang baik dan benar.
2. Identitas per halaman harus jelas (misalnya : halaman, kode kuliah).

Penilaian:

1. **Bagian 1** : Laporan (25%)
 - a. Langkah penyelesaian masalah setiap fitur (10%)
 - b. Hasil pengujian dan analisis algoritma (10%)
 - c. Komponen-komponen lain dalam laporan (5%)
2. **Bagian 2** : Implementasi Program (75%)
 - a. Kebenaran program (30%)
 - b. Pemahaman terhadap cara kerja program (25%)
 - c. Interface, Features, dan Unsur Kreativitas (20%)
3. **Bagian 3** : Bonus (15%)
 - a. Melakukan *deployment* ke dalam website
 - b. Membuat video demonstrasi program
 - c. Membuat fitur rekomendasi kata apabila terjadi kesalahan

Arsip pengumpulan :

- Source program yang bisa dijalankan disertai *readme.txt*
- Laporan (*soft copy*)

--- Selamat Mengerjakan, *it's not worth it if you're not having fun* ---