

Tugas Besar III IF2211 Strategi Algoritma
Deteksi *Spam* pada Media Sosial atau *Chat-Messenger* dengan Algoritma Pencocokan String

Batas pengumpulan : Senin, 23 April 2017 jam 13.

Arsip pengumpulan : - CD yang berisi Source dan Exe program disertai readme.txt
- Laporan (hard copy)

Tempat pengumpulan : Di atas loker Lab IRK

Deskripsi Tugas:

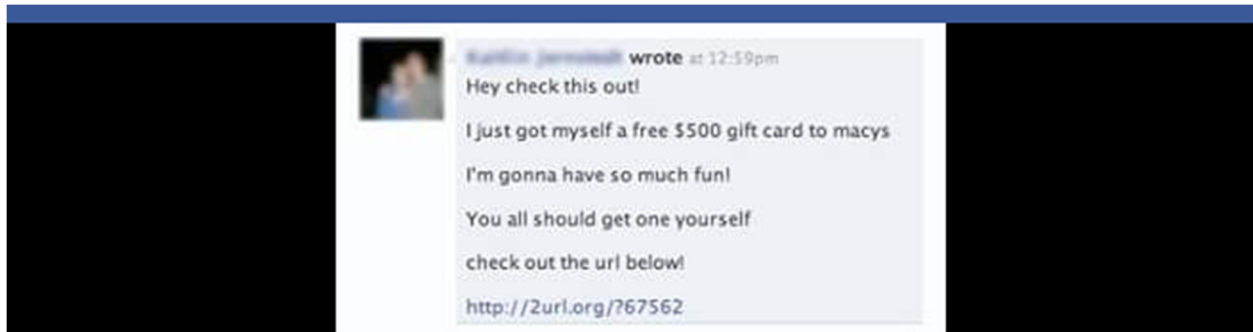


Electronic-spam merupakan pesan elektronik yang tidak diinginkan penerimanya, bisa dalam bentuk surat elektronik, SMS, *posting* atau komentar di media sosial yang muncul di *timeline* kita, ataupun pesan pada *chat-messenger*. *Spammer* melakukan *spamming* untuk tujuan tertentu, paling banyak untuk menyebarkan iklan. Penentuan *spam* sangat bersifat subjektif, artinya *spam* untuk kita, belum tentu *spam* untuk pengguna lain. Gambar 1-2 merupakan contoh *spam* pada *twitter*, *facebook*, dan *line*.



<https://econsultancy.com/blog/4218-the-seven-twitter-sins-a-comprehensive-list-of-twitter-spam-techniques>

Gambar 1. Contoh spam pada media sosial *twitter*:
(a) hashtag/trending topic spam; (b) autoresponder spam



Facebook Security

This is an example of a spam Wall post made from the account of a compromised user. Notice the "Hey check this out!" line and the enticing offer for a free Macy's gift card.

Examples of Spam · Oct 30, 2008 ·

<https://m.facebook.com/security/photos/a.36604031885.58084.31987371885/36604061885/?type=3&source=43>

Gambar 2. Contoh *spam* pada media sosial *facebook*

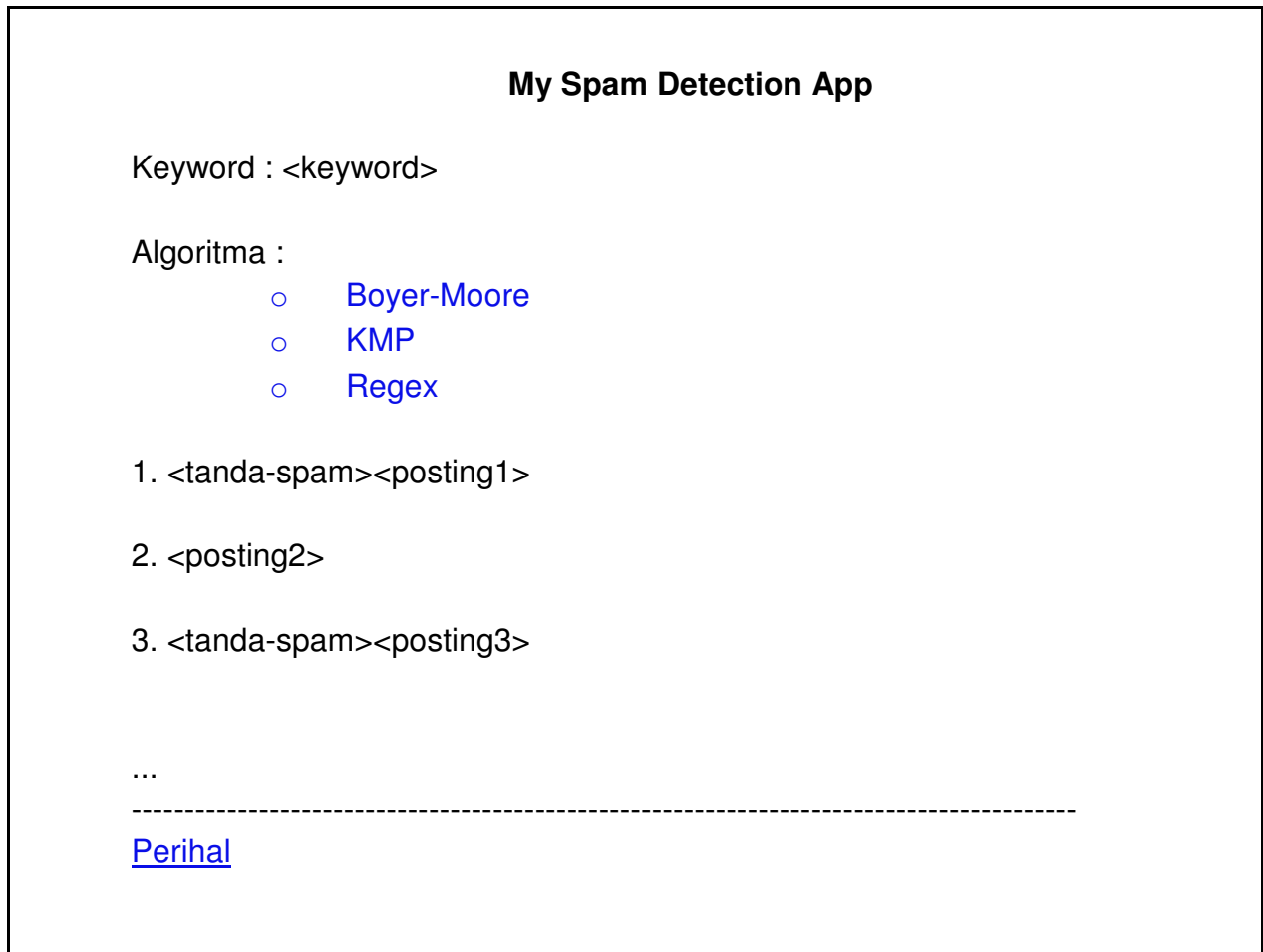
Algoritma pencocokan string (*pattern*) Knuth-Morris-Pratt (KMP) dan Algoritma Boyer-Moore merupakan algoritma yang lebih baik daripada *brute force*. Pada Tugas Besar III kali ini Anda diminta membuat aplikasi sederhana deteksi *spam* pada media sosial dengan kedua algoritma tersebut, plus menggunakan *regular expression (regex)*. Teks yang akan Anda proses adalah *posting* berbahasa Indonesia. Pengguna aplikasi ini akan memberikan masukan berupa *keyword* spam, dan menandai daftar *posting* yang dikategorikan *spam* terurut berdasarkan tanggal.

Pencocokan string yang anda buat adalah *exact matching* (untuk KMP dan BM) jadi *posting* yang diproses mengandung string yang tepat sama dengan keyword spam dari pengguna. Sedangkan bila menggunakan *regex* maka tidak selalu *exact matching*. Pencarian juga tidak bersifat *case sensitive*, jadi huruf besar dan huruf kecil dianggap sama (hal ini dapat dilakukan dengan menganggap seluruh karakter di dalam pattern dan teks sebagai huruf kecil semua atau huruf kapital semua).

Kumpulan posting diambil secara otomatis menggunakan *Facebook API* atau *Twitter API* (<https://developer.twitter.com/en/docs/tweets/search/overview>) atau *Line API* (<https://developers.line.me/en/services/messaging-api/>) atau api dari media sosial lainnya.

Spesifikasi program:

1. Aplikasi deteksi spam yang anda buat merupakan aplikasi berbasis web yang menerima *keyword* pencarian, misalnya “klik link ini”. Tampilan antarmuka pengguna-komputer kira-kira seperti Gambar 3 di bawah ini:



Gambar 3. Contoh antarmuka aplikasi sederhana deteksi spam

Perihal: pranala ke halaman tentang program dan pembuatnya. Anda dapat menambahkan menu lainnya, gambar, logo, dan sebagainya

2. Deteksi *spam* menggunakan hasil implementasi algoritma KMP, Boyer-Moore, dan Regex dengan menggunakan Bahasa C#, Java, Python, Golang. Pencocokan string dilakukan pada konten *posting*.

Data Uji

Data uji yang digunakan dapat anda tentukan sendiri, minimal terdapat 20 posting yang diproses. Posting dapat berbahasa Indonesia atau Inggris.

Lain – lain:

1. Anda dapat menambahkan fitur fungsional lain yang menunjang program yang anda buat (unsur kreatifitas diperbolehkan/dianjurkan).
2. Program berbasis web dan dapat dikembangkan dengan salah satu kakas: PHP, JSP (Java Server Pages), atau ASP.
3. Program implementasi Boyer-Moore dan KMP menggunakan salah satu dari Bahasa C#, Java, Python, Golang.
4. Akses konten media sosial dapat menggunakan salah satu API yang tersedia.
5. Tugas dikerjakan per kelompok dengan jumlah anggota maksimal 3 orang dan tidak boleh sama dengan anggota kelompok sebelumnya.
6. Program harus modular dan mengandung komentar yang jelas.
7. Mahasiswa harus membuat program sendiri kecuali library XML dan HTML *parser*, tetapi belajar dari contoh-contoh program serupa yang sudah ada tidak dilarang (tidak boleh mengkopi *source code* dari program orang lain).
8. Keterlambatan pengumpulan akan mengurangi nilai.
9. Program disimpan di dalam folder StrAlgo3-xxxxx. Lima digit terakhir adalah NIM anggota terkecil. Di dalam folder tersebut terdapat tiga folder bin, src dan doc yang masing-masing berisi :
 - a. Folder bin berisi executable file (exe)
 - b. Folder src berisi source code dari program
 - c. Folder test berisi data uji.
 - d. Folder doc berisi dokumentasi program dan readme

Folder ini disimpan dalam bentuk CD untuk dikumpulkan bersama berkas laporan dimasukkan kedalam amplop coklat.

10. Semua pertanyaan menyangkut tugas ini harus dikomunikasikan melalui milis agar dapat dicermati oleh semua peserta kuliah IF2211 (milis IF2211@students.if.itb.ac.id).
11. Demo program akan dilaksanakan pada tanggal yang dimumkan oleh asisten. Peserta mengisi jadwal demo yang disediakan pada saat pengumpulan tugas.
12. Tiap anggota harus memahami proses pembuatan program, karena akan ada pertanyaan-pertanyaan yang harus dijawab per individu.

13. Pada saat demo, asisten akan memanggil per kelompok sesuai jadwal yang telah diisi sebelumnya. Kelompok yang tidak berkepentingan dilarang masuk. Demo dilakukan di Lab IRK.
14. Bonus (nilai maksimal 10): Setiap kelompok membuat video aplikasi yang mereka buat kemudian mengunggahnya ke Youtube. Pada waktu demo aplikasi di depan asisten, mahasiswa mengakses video Youtube tersebut dan memutarinya di depan asisten sebelum memulai demo.

Isi laporan :

Cover: Cover laporan ada foto anggota kelompok (foto bertiga) yang paling keren. Foto ini menggantikan logo “gajah” ganesha.

Bab 1: Deskripsi masalah (dapat meng-copy paste file tugas ini)

Bab 2: Dasar teori (ringkas)

Bab 3: Analisis Pemecahan Masalah. Langkah-langkah pemecahan masalah ada di sini beserta contoh ilustrasi.

Bab 4: Implementasi dan pengujian. Bab ini berisi:

- a. Spesifikasi teknis program, termasuk di dalamnya struktur data atau kelas objek yang didefinisikan, fungsi dan prosedur (header fungsi dan prosedur saja, tidak perlu source code), antarmuka, dan lain-lain yang dianggap perlu.
- b. Eksperimen/pengujian dengan contoh-contoh query.
- c. Analisis hasil pengujian.

Bab 5: Kesimpulan dan saran (hasil yang dicapai, saran pengembangan). Serta kesan/komentar tentang tugas ini.

Tuliskan juga referensi (buku, web), yang dipakai/diacu di dalam Daftar Referensi.

Keterangan laporan:

- a. Laporan ditulis dalam bahasa Indonesia yang baik dan benar, tidak perlu panjang tetapi tepat sasaran dan jelas.
- b. Laporan tidak perlu memakai cover mika dan dijilid. Cukup dibuat agar laporan tidak akan tercecer bila dibaca.
- c. Laporan boleh menggunakan kertas rius, boleh bolak-balik, boleh dalam satu halaman kertas terdapat dua halaman tulisan asalkan masih terbaca.
- d. Identitas per halaman harus jelas (misalnya : halaman, kode kuliah).

Penilaian:

- a. Kebenaran program (40%): program mampu berjalan sesuai dengan spesifikasi yang diberikan.
- b. Demo – pemahaman Anda dalam pembuatan program (30%)
- c. Laporan (20%)
- d. Interface, feature-feature program, dan unsur kreativitas (20%)

-selamat mengerjakan-