

LAPORAN TUGAS KECIL 4

IF2211 Strategi Algoritma

**Ekstraksi Informasi dari Artikel Berita dengan Algoritma
Pencocokan String**



Dipersiapkan oleh:

Morgen Sudyanto

13518093

**Sekolah Teknik Elektro dan Informatika – Institut Teknologi
Bandung**

Jl. Ganesha 10, Bandung 40132

BAB I

DASAR TEORI

I. Algoritma Knuth-Morris-Pratt (KMP)

KMP adalah sebuah algoritma pencocokan string yang melakukan pencarian pola dari kiri ke kanan seperti algoritma brute force, tetapi dengan pergeseran yang dilakukan secara lebih ‘pintar’.

Pada algoritma bruteforce, pergeseran dilakukan satu per satu. Ini menyebabkan pencarian kurang efisien, karena algoritma mengecek lagi string yang sudah di pernah dicek sebelumnya. Algoritma KMP melakukan pergeseran ini dengan jauh lebih efisien.

Misalkan ada suatu teks T dan pola P. Jika ketidakcocokan terjadi pada $T[i]$ dan $P[j]$, maka pergeseran yang dilakukan adalah sebesar prefix terbesar $P[0..j-1]$ yang juga merupakan suffix dari $P[1..j-1]$.

Untuk dapat menjalankan algoritma KMP, diperlukan sebuah preprocessing untuk membentuk fungsi pinggiran KMP (border function / fail function). Misalkan j adalah posisi terjadinya ketidakcocokan dan k adalah satu posisi sebelum terjadinya ketidakcocokan ($j-1$), maka fungsi pinggiran KMP ($fail[k]$) didefinisikan sebagai panjang prefix terpanjang dari $P[0..k]$ yang juga merupakan suffix dari $P[1..k]$.

Contoh dari pembuatan border function:

P = “kukukakiku”

j	0	1	2	3	4	5	6	7	8	9
P[j]	k	u	k	u	k	a	k	i	k	u
k	-	0	1	2	3	4	5	6	7	8
fail[k]	-	0	1	2	3	0	1	0	1	2

Tabel 1.1 Pembuatan border function untuk P = “kukukakiku”

Saat $j = 5$, k bernilai 4. Dapat dilihat bahwa $P[0..4]$ adalah “kukuk” dan $P[1..4]$ adalah “ukuk”. Prefix terpanjang yang juga merupakan suffix adalah “kuk”, yang memiliki panjang 3. Maka, nilai dari $fail[4]$ adalah 3.

Algoritma KMP lalu hanya perlu melihat tabel border function untuk mengetahui berapa banyak ia harus bergeser. Jika terjadi ketidakcocokan pada $T[i]$ dan $P[j]$, maka j diset dengan nilai $fail[j-1]$.

II. Algoritma Boyer-Moore

BM adalah sebuah algoritma pencocokan string yang memanfaatkan dua buah teknik, yaitu teknik “looking-glass”, yang mencari pola P pada suatu teks T dari bagian akhir string ke bagian awal string, dan teknik “character-jump”, yang melakukan suatu lompatan ketika $T[i] = x$ dan terjadi ketidakcocokan pada $T[i]$ dan $P[j]$ berdasarkan beberapa kasus yang mungkin terjadi.

Kasus pertama yaitu ketika P mengandung x . P akan digeser agar kemunculan terakhir x sejajar dengan $T[i]$. Kasus kedua yaitu ketika P mengandung x tetapi pergeseran tidak mungkin dilakukan. P akan digeser sejauh 1, $P[j]$ menjadi sejajar dengan $T[i+1]$. Kasus ketiga, jika kasus pertama dan kedua tidak cocok (P tidak mengandung x), maka P digeser agar $P[0]$ sejajar dengan $T[i+1]$.

Preprocessing yang dilakukan oleh algoritma BM adalah preprocessing untuk membentuk fungsi kemunculan terakhir. Misalkan x adalah sebuah huruf yang sudah diubah menjadi angka (nilai asciinya). $last[x]$ didefinisikan sebagai index terbesar i dimana $P[i] = x$, atau -1 bila tidak ada index yang memenuhi.

Contoh dari pembuatan last occurrence function:

$P = \text{“aaabacada”}$

x	a	b	c	d	e
last[x]	8	3	5	7	-1

Tabel 1.2 Pembuatan last occurrence function untuk $P = \text{“aaabacada”}$

III. Regex

Regular expression (regex) adalah notasi standar yang mendeskripsikan suatu pola (pattern) berupa urutan karakter atau string. Regex digunakan untuk pencocokan string (string matching) dengan efisien. Python sendiri memiliki library regex, yaitu `re`.

Beberapa fungsi yang ada di dalam library `re` Python antara lain:

- `re.sub(pattern, replace, string)`: Mensubstitusikan pattern yang ada di dalam string dengan replace.

Contoh: `re.sub('config', '', 'coconfigfig')` mengembalikan `'config'`

- `re.match(pattern, string)`: Mencocokkan bagian awal string (prefix) dengan pattern. Mengembalikan match object bila ada 0 atau lebih karakter pattern yang cocok pada bagian awal string.
Contoh: `re.match('pola', 'akupola')` mengembalikan object None karena pola bukan merupakan bagian awal string.
- `re.search(pattern, string)`: Mencocokkan seluruh bagian string dengan pattern untuk mencari lokasi pertama dimana suatu kecocokan terjadi. Mengembalikan match object bila ada 0 atau lebih karakter pattern yang cocok pada suatu bagian string.
Contoh: `re.search('pola', 'akupola')` mengembalikan match object karena pola merupakan bagian dari 'akupola'.

BAB II

KODE PROGRAM

Untuk backend, program memanfaatkan kaskas flask. Sedangkan untuk frontend, program memanfaatkan CSS dan JS, dibantu dengan framework Bootstrap. Algoritma KMP, BM dan Regex diimplementasikan dalam bahasa Python.

1. algo.py (Algoritma KMP dan BM)

```
def computeFail(pattern):
    m = len(pattern)
    fail = [0 for x in range(m)]
    i, j = 1, 0
    while i < m:
        if pattern[j].lower() == pattern[i].lower():
            fail[i] = j + 1
            i += 1
            j += 1
        elif j > 0:
            j = fail[j-1]
        else:
            fail[i] = 0
            i += 1
    return fail

def KMP(text, pattern):
    n, m = len(text), len(pattern)
    fail = computeFail(pattern)
    i, j = 0, 0
    while i < n:
        if pattern[j].lower() == text[i].lower():
            if j == m - 1:
                return i - m + 1
            i += 1
            j += 1
        elif j > 0:
            j = fail[j-1]
        else:
            i += 1
    return -1

def buildLast(pattern):
    last = [-1 for x in range(128)]
```

```

        for i in range(len(pattern)):
            last[ord(pattern[i].lower())] = i
        return last

def BM(text, pattern):
    last = buildLast(pattern)
    n, m = len(text), len(pattern)
    i = m - 1
    if i > n - 1:
        return -1
    j = m - 1
    while True:
        if (pattern[j].lower() == text[i].lower()):
            if (j == 0):
                return i
            else:
                i -= 1
                j -= 1
        else:
            lo = last[ord(text[i].lower())]
            i += m - min(j, lo + 1)
            j = m - 1
    if (i > n - 1):
        break
    return -1

```

2. app.py (Backend)

```

import re, os, random, string, json, nltk
from nltk.tokenize import sent_tokenize
from algo import BM, KMP
from flask import Flask, render_template,
render_template_string, request, redirect

app = Flask(__name__)
app.config['SECRET_CODE'] = 'v3rys3cr3tstring'

# Initialize results directory
results_dir = os.path.join(app.instance_path, 'results')
if not os.path.exists(os.path.dirname(results_dir)):
    os.makedirs(results_dir)

nltk.download('punkt')

```

```

keyword = ''
algorithm = ''
fileNames = []
foundSentences = []
foundCounts = []
foundDates = []
code = ''

@app.errorhandler(404)
def page_not_found(e):
    return render_template('404.html'), 404

# Home page
@app.route('/')
def home():
    return render_template('index.html')

# Displays result
@app.route('/result/<fileCode>')
def result(fileCode):
    global keyword, algorithm, code

    # For challenge purposes
    if fileCode == 'v3rys3cr3tstring':
        return render_template('congrats.html')
    if not os.path.isfile(results_dir + '/' + fileCode):
        return render_template_string(fileCode)

    data = json.load(open(results_dir + '/' + fileCode))
    data = data[0]
    keyword = data['keyword']
    algorithm = data['algorithm']
    fileNames = data['fileNames']
    foundSentences = data['foundSentences']
    foundCounts = data['foundCounts']
    foundDates = data['foundDates']
    code = data['code']
    return render_template('result.html', keyword = keyword,
algorithm = algorithm, fileNames = fileNames, foundSentences =
foundSentences, foundCounts = foundCounts, foundDates =
foundDates, code = code)

# Text input

```

```

@app.route('/upload/text', methods=['POST'])
def uploadText():
    global keyword, algorithm, code
    clear()
    keyword = request.form['keywordText']
    algorithm = request.form['algorithmText']
    inputText = request.form['inputText']

    fileNames.append('Text')
    sentences = sent_tokenize(inputText)
    foundSentence, foundCount, foundDate =
findSentence(sentences, algorithm, keyword)
    foundSentences.append(foundSentence)
    foundCounts.append(foundCount)
    foundDates.append(foundDate)
    code = ''.join(random.choices(string.ascii_letters +
string.digits, k=16))
    data = [{'keyword': keyword, 'algorithm': algorithm,
'fileNames': fileNames, 'foundSentences': foundSentences,
'foundCounts': foundCounts, 'foundDates': foundDates, 'code':
code}]
    with open(results_dir + '/' + code, 'w') as file:
        json.dump(data, file)
    return redirect('/result/' + code)

# File input
@app.route('/upload/file', methods=['POST'])
def uploadFile():
    global keyword, algorithm, code
    clear()
    keyword = request.form['keywordFile']
    algorithm = request.form['algorithmFile']
    inputFiles = request.files.getlist('inputFiles')
    for file in inputFiles:
        fileNames.append(file.filename)
        text = file.read().decode('utf-8')
        sentences = sent_tokenize(text)
        foundSentence, foundCount, foundDate =
findSentence(sentences, algorithm, keyword)
        foundSentences.append(foundSentence)
        foundCounts.append(foundCount)
        foundDates.append(foundDate)
    code = ''.join(random.choices(string.ascii_letters +
string.digits, k=16))

```



```

        data = [{'keyword': keyword, 'algorithm': algorithm,
'fileNames': fileNames, 'foundSentences': foundSentences,
'foundCounts': foundCounts, 'foundDates': foundDates, 'code':
code}]
        with open(results_dir + '/' + code, 'w') as file:
            json.dump(data, file)
        return redirect('/result/' + code)

# See previous result by using given code
@app.route('/upload/code', methods=['POST'])
def uploadCode():
    return redirect('/result/' + request.form['code'])

# Clear variables
def clear():
    global keyword, algorithm, code
    keyword = ''
    algorithm = ''
    code = ''
    fileNames.clear()
    foundSentences.clear()
    foundCounts.clear()
    foundDates.clear()

# Find sentence(s) that contains keyword using chosen algorithm
def findSentence(sentences, algorithm, keyword):
    result = []
    count = []
    date = []
    newsDate = '-'
    # Find date from any sentence, starting from top (news
title)
    for sentence in sentences:
        newsDate = findDate(sentence, '')
        if newsDate != '-':
            break
    for sentence in sentences:
        if algorithm == 'KMP':
            index = KMP(sentence, keyword)
            if index != -1:
                result.append(highlight(sentence, keyword))
                count.append(findCount(sentence, keyword))
                date.append(findDate(sentence, keyword))
                if date[-1] == '-':

```

```

        date[-1] = newsDate
    elif algorithm == 'BM':
        index = BM(sentence, keyword)
        if index != -1:
            result.append(highlight(sentence, keyword))
            count.append(findCount(sentence, keyword))
            date.append(findDate(sentence, keyword))
            if date[-1] == '-':
                date[-1] = newsDate
        else:
            if re.search(keyword, sentence, flags=re.IGNORECASE):
                result.append(highlight(sentence, keyword))
                count.append(findCount(sentence, keyword))
                date.append(findDate(sentence, keyword))
                if date[-1] == '-':
                    date[-1] = newsDate
    return result, count, date

# Highlight keyword that is found in the sentence
def highlight(sentence, keyword):
    bold = "<b>" + keyword + "</b>"
    return re.sub(keyword, bold, sentence,
flags=re.IGNORECASE)

# Returns the closest number to the given keyword.
def findCount(sentence, keyword):
    regex =
re.search('(?:^|\s)((?:\d*\s)*\d+)\s(?:[\\(\-\\s\\w,:])*?' +
keyword + '|' + keyword +
'(?:[\\(\-\\s\\w,:])*?\s((?:\d*\s)*\d+)(?:$|\s|.|,))', sentence,
flags=re.IGNORECASE)
    if regex is not None:
        if regex.group(1) is not None:
            return regex.group(1)
        if regex.group(2) is not None:
            return regex.group(2)
    return '-'

# Returns the closest date to the given keyword.
# Example of supported date formats:
# 24 April
# 4 Mei
# 24/04/2020
# 17/04

```

```
# 2/12/19
def findDate(sentence, keyword):
    # DD Bulan
    regex =
re.search('(\d\d?\s(?:Januari|Februari|Maret|April|Mei|Juni|Jul
i|Agustus|September|November|Desember|Jan|Feb|Mar|Apr|Mei|Jun|J
ul|Aug|Sep|Okt|Nov|Des)).+(?:[\\(\-\\s\\w,":])*?' + keyword + '|'
+ keyword +
'(?:[\\(\-\\s\\w,":])*?(\d\d?\s(?:Januari|Februari|Maret|April|Mei
|Juni|Juli|Agustus|September|November|Desember|Jan|Feb|Mar|Apr|
Mei|Jun|Jul|Aug|Sep|Okt|Nov|Des)).+', sentence,
flags=re.IGNORECASE)
    if regex is not None:
        if regex.group(1) is not None:
            return regex.group(1)
        if regex.group(2) is not None:
            return regex.group(2)

    # DD/MM/YYYY
    regex =
re.search('(?:\d\d?\/\d\d?\/?\d\d?\d?\d?)\)??.+(?:[\\(\-\\s\\w,":
])*?' + keyword + '|' + keyword +
'(?:[\\(\-\\s\\w,":])*?(?:\d\d?\/\d\d?\/?\d\d?\d?\d?)\)??.+',
sentence, flags=re.IGNORECASE)
    if regex is not None:
        if regex.group(1) is not None:
            return regex.group(1)
        if regex.group(2) is not None:
            return regex.group(2)
    return '-'
```

3. index.html (Frontend - homepage)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <!-- Bootstrap css -->
    <link rel="stylesheet"
```

```
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
```

```
<!-- Font Awesome -->
<script
src="https://kit.fontawesome.com/be39655df2.js"></script>
```

```
<!-- Own stylesheet -->
<link rel="stylesheet" href="{ { url_for('static',
filename='styles.css') }}">
```

```
<title>CoroFind</title>
</head>
```

```
<body class="background-grey">
  <nav class="navbar navbar-dark bg-primary">
    <a class="navbar-brand" href="/">CoroFind</a>
  </nav>
  <div class="jumbotron">
    <h1 class="display-4">Welcome</h1>
    <p class="lead">This is a website that helps you extract
information from news websites.</p>
    <hr class="my-4">
    <p>Start by clicking on one of these boxes below:</p>
```

```
  <div id="accordion">
    <div class="card">
      <div class="card-header" id="headingText">
        <h5 class="mb-0">
          <button class="btn btn-link"
data-toggle="collapse" data-target="#collapseText"
aria-expanded="true" aria-controls="collapseText">
            Search by pasting text
          </button>
        </h5>
      </div>
      <div id="collapseText" class="collapse"
aria-labelledby="headingText" data-parent="#accordion">
        <div class="card-body">
          <form action="/upload/text"
method="post" enctype="multipart/form-data">
            <div class="form-group">
              <label for="inputText">Input your
```

```

text:</label>
                                <textarea class="form-control"
rows="3" name="inputText" id="inputText" required></textarea>
                                </div>
                                <div class="form-group">
                                <label for="keywordText">Keyword:
</label>
                                <input class="form-control"
type="text" name="keywordText" id="keywordText"
placeholder="ODP" required>
                                </div>
                                <label for="algorithmText">Choose an
algorithm: </label>
                                <div class="form-check">
                                <input class="form-check-input"
type="radio" name="algorithmText" id="KMP" value="KMP" checked>
                                <label class="form-check-label"
for="KMP">KMP</label>
                                </div>
                                <div class="form-check">
                                <input class="form-check-input"
type="radio" name="algorithmText" id="BM" value="BM">
                                <label class="form-check-label"
for="BM">BM</label>
                                </div>
                                <div class="form-check">
                                <input class="form-check-input"
type="radio" name="algorithmText" id="Regex" value="Regex">
                                <label class="form-check-label"
for="Regex">Regex</label>
                                </div>
                                <button class="btn btn-primary mt-3"
type="submit">Submit</button>
                                </form>
                                </div>
                                </div>
                                </div>
                                <div class="card">
                                <div class="card-header" id="headingFile">
                                <h5 class="mb-0">
                                <button class="btn btn-link"
data-toggle="collapse" data-target="#collapseFile"
aria-expanded="false" aria-controls="collapseFile">

```



```

for="Regex">Regex</label>
                                </div>
                                <button class="btn btn-primary mt-3"
type="submit">Submit</button>
                                </form>
                            </div>
                        </div>
                    </div>
                </div>

                <div class="card">
                    <div class="card-header" id="headingCode">
                        <h5 class="mb-0">
                            <button class="btn btn-link"
data-toggle="collapse" data-target="#collapseCode"
aria-expanded="false" aria-controls="collapseCode">
                                Open with code
                            </button>
                        </h5>
                    </div>
                    <div id="collapseCode" class="collapse"
aria-labelledby="headingCode" data-parent="#accordion">
                        <div class="card-body">
                            <form action="/upload/code"
method="post" enctype="multipart/form-data">
                                <div class="form-group">
                                    <label for="inputCode">Enter your
code: </label>
                                    <input class="form-control"
type="text" name="code" id="code" placeholder="Alphanumeric
characters">
                                </div>
                                <button class="btn btn-primary"
type="submit">Submit</button>
                            </form>
                        </div>
                    </div>
                </div>
            </div>
        </div>
        <div class="navbar fixed-bottom navbar-dark
bg-primary">
            <span class="smaller-text white-text">Made by Morgen
Sudyanto (13518093) as a part of 'Algorithm Strategy' mini
project.</span>

```

```

        <ul class="navbar-nav navbar-expand-md">
            <li class="nav-item">
                <a href="https://twitter.com/moondemon68"><i
class="footer-icon fab fa-twitter"></i></a>
            </li>
            <li class="nav-item">
                <a href="https://github.com/moondemon68"><i
class="footer-icon fab fa-github"></i></a>
            </li>
            <li class="nav-item">
                <a
href="https://www.instagram.com/morgen_68/"><i
class="footer-icon fab fa-instagram"></i></a>
            </li>
            <li class="nav-item">
                <a href="mailto:moondemon68@gmail.com"><i
class="footer-icon fas fa-envelope"></i></a>
            </li>
        </ul>
    </footer>

    <!-- Bootstrap js -->
    <script
src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4Y
fRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMga3JDZ
wrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/boot
strap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGz
IxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

    <!-- Custom js -->
    <script>
    $(document).on('change', '.custom-file-input', (e) => {

        const $this = $(e.target),
        $label = $this.prev('label'),
        $files = $this[0].files;
    
```



```

        let fileName = '';

        if ($files && $files.length > 1)
            fileName = ($this.attr('data-multiple-target') ||
            '').replace('{count}', $files.length);
        else if (e.target.value)
            fileName = e.target.value.split('\\').pop();

        if (fileName) {
            $label.find('.span-choose-file').html(fileName);
        } else {
            $label.html($label.html());
        }

    });
</script>
</body>
</html>

```

4. result.html (Frontend - search results)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <!-- Bootstrap css -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bo
otstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0
hcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

    <!-- Font Awesome -->
    <script src="https://kit.fontawesome.com/25734e3e8d.js"
crossorigin="anonymous"></script>

    <!-- Own stylesheet -->
    <link rel="stylesheet" href="{ url_for('static',
filename='styles.css') }}">

```

```

        <title>Results - CoroFind</title>
</head>
<body class="background-grey">
    <nav class="navbar navbar-dark bg-primary">
        <a class="navbar-brand" href="/">CoroFind</a>
    </nav>
    <div class="container-fluid container-margin">
        {% if keyword | length %}
            <div class="row m-2">
                <div class="col">You searched for <b>{{ keyword
}}</b>.</div>
            </div>
        {% endif %}
        {% if algorithm | length %}
            <div class="row m-2">
                <div class="col">Algorithm used: {{ algorithm
}}</div>
            </div>
        {% endif %}
        {% for i in range(fileNames | length) %}
            <div class="card m-4">
                <div class="card-header">
                    <button class="btn btn-link btn-lg"
data-toggle="collapse" data-target="#collapse-{{ i }}"
aria-expanded="true" aria-controls="collapseOne">{{
fileNames[i] }}</button>
                </div>
                <ul class="list-group list-group-flush collapse
show" id="collapse-{{ i }}">
                    {% for j in range(foundSentences[i] |
length) %}
                        <li class="list-group-item">
                            <div class="row m-3">
                                <div class="col-md-6
col-sm-12">Jumlah: {{ foundCounts[i][j] }}</div>
                                <div class="col-md-6
col-sm-12">Tanggal: {{ foundDates[i][j] }}</div>
                            </div>
                            <div class="row m-3">
                                <div class="col">{{
foundSentences[i][j] | safe }}</div>
                            </div>
                        </li>

```

```

        {% endfor%}
    </ul>
</div>
{% endfor %}
<div class="row m-2">
    <div class="col">Use this code to view this page
again: {{ code }}</div>
</div>
<div class="row m-2">
    <div class="col"><a href="/" class="btn
btn-primary">Click here to go back</a></div>
</div>
</div>

<footer class="navbar fixed-bottom navbar-dark
bg-primary">
    <span class="smaller-text white-text">Made by Morgen
Sudyanto (13518093) as a part of 'Algorithm Strategy' mini
project.</span>
    <ul class="navbar-nav navbar-expand-md">
        <li class="nav-item">
            <a href="https://twitter.com/moondemon68"><i
class="footer-icon fab fa-twitter"></i></a>
        </li>
        <li class="nav-item">
            <a href="https://github.com/moondemon68"><i
class="footer-icon fab fa-github"></i></a>
        </li>
        <li class="nav-item">
            <a
href="https://www.instagram.com/morgen_68/"><i
class="footer-icon fab fa-instagram"></i></a>
        </li>
        <li class="nav-item">
            <a href="mailto:moondemon68@gmail.com"><i
class="footer-icon fas fa-envelope"></i></a>
        </li>
    </ul>
</footer>

<!-- Bootstrap js -->
<script
src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4Y

```

```
fRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
d/popper.min.js"
integrity="sha384-U02eT0CpHqdSJJQ6hJty5KVphtPhzWj9W01c1HTMGa3JDZ
wrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/boot
strap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGz
IxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>
</body>
</html>
```

5. styles.css (CSS)

```
/*Footer*/
footer {
    line-height: 3rem;
}

.container-margin {
    margin-top: 2%;
    margin-bottom: 5%;
}

.footer-icon {
    padding: 10px;
    color: #FFFFFF;
}

.fa-twitter:hover {
    color: #007BFF;
}

.fa-github:hover {
    color: #000000;
}

.fa-instagram:hover {
    color: #CD4BA1;
}
```

```

.fa-envelope:hover {
    color: #D93025;
}

.background-grey {
    background-color: #E9ECEF;
}

.smaller-text {
    font-size: 1rem;
}

.white-text {
    color: #FFFFFF;
}

@media only screen and (max-width: 767px) {
    .navbar {
        position: relative;
    }
}

```

6. 404.html (error handler)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <!-- Bootstrap css -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bo
otstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0
hcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

    <!-- Font Awesome -->
    <script
src="https://kit.fontawesome.com/be39655df2.js"></script>

```

```

    <!-- Own stylesheet -->
    <link rel="stylesheet" href="{{ url_for('static',
filename='styles.css') }}">

    <title>CoroFind</title>
</head>
<body class="background-grey">
    <nav class="navbar navbar-dark bg-primary">
    <a class="navbar-brand" href="/">CoroFind</a>
    </nav>
    <p style="text-align:center;"></p>
    <h1 style="text-align:center;">Page not found lmao</h1>
    <p style="text-align:center; font-size:0.5rem">Imagine
getting rickrolled in 2020</p>
    <p style="text-align:center;"><iframe
src="https://giphy.com/embed/lgcUUCXgC8mEo" width="480"
height="360" frameborder="0" class="giphy-embed"
allowFullScreen></iframe><p><a
href="https://giphy.com/gifs/rick-roll-lgcUUCXgC8mEo"></a></p><
/p>
    <p style="text-align:center;"><a href="/" class="btn
btn-primary">Click here to go back</a></p>
    <footer class="navbar fixed-bottom navbar-dark
bg-primary">
    <span class="smaller-text white-text">Made by Morgen
Sudyanto (13518093) as a part of 'Algorithm Strategy' mini
project.</span>
    <ul class="navbar-nav navbar-expand-md">
        <li class="nav-item">
            <a href="https://twitter.com/moondemon68"><i
class="footer-icon fab fa-twitter"></i></a>
        </li>
        <li class="nav-item">
            <a href="https://github.com/moondemon68"><i
class="footer-icon fab fa-github"></i></a>
        </li>
        <li class="nav-item">
            <a
href="https://www.instagram.com/morgen_68/"><i
class="footer-icon fab fa-instagram"></i></a>
        </li>
        <li class="nav-item">
            <a href="mailto:moondemon68@gmail.com"><i

```

```

class="footer-icon fas fa-envelope"></i></a>
    </li>
</ul>
</footer>

<!-- Bootstrap js -->
<script
src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4Y
fRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMga3JDZ
wrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/boot
strap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGz
IxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

<!-- Custom js -->
<script>
$(document).on('change', '.custom-file-input', (e) => {

    const $this = $(e.target),
    $label = $this.prev('label'),
    $files = $this[0].files;
    let fileName = '';

    if ($files && $files.length > 1)
        fileName = ($this.attr('data-multiple-target') ||
'').replace('{count}', $files.length);
    else if (e.target.value)
        fileName = e.target.value.split('\\').pop();

    if (fileName) {
        $label.find('.span-choose-file').html(fileName);
    } else {
        $label.html($label.html());
    }

});
</script>

```

```
</body>
</html>
```

7. congrats.html (Page tambahan sebagai challenge)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">

  <!-- Bootstrap css -->
  <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bo
otstrap.min.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0
hcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

  <!-- Font Awesome -->
  <script
src="https://kit.fontawesome.com/be39655df2.js"></script>

  <!-- Own stylesheet -->
  <link rel="stylesheet" href="{{ url_for('static',
filename='styles.css') }}">

  <title>CoroFind</title>
</head>
<body class="background-grey">
  <nav class="navbar navbar-dark bg-primary">
    <a class="navbar-brand" href="/">CoroFind</a>
  </nav>
  <h1 style="text-align:center;">Congratulations!</h1>
  <p style="text-align:center;"></p>
  <footer class="navbar fixed-bottom navbar-dark
bg-primary">
    <span class="smaller-text white-text">Made by Morgen
Sudyanto (13518093) as a part of 'Algorithm Strategy' mini
project.</span>
```



```

        <ul class="navbar-nav navbar-expand-md">
            <li class="nav-item">
                <a href="https://twitter.com/moondemon68"><i
class="footer-icon fab fa-twitter"></i></a>
            </li>
            <li class="nav-item">
                <a href="https://github.com/moondemon68"><i
class="footer-icon fab fa-github"></i></a>
            </li>
            <li class="nav-item">
                <a
href="https://www.instagram.com/morgen_68/"><i
class="footer-icon fab fa-instagram"></i></a>
            </li>
            <li class="nav-item">
                <a href="mailto:moondemon68@gmail.com"><i
class="footer-icon fas fa-envelope"></i></a>
            </li>
        </ul>
    </footer>

    <!-- Bootstrap js -->
    <script
src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4Y
fRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd
/popper.min.js"
integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMGa3JDZ
wrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/boot
strap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJGz
IxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

    <!-- Custom js -->
    <script>
    $(document).on('change', '.custom-file-input', (e) => {

        const $this = $(e.target),
        $label = $this.prev('label'),
        $files = $this[0].files;
    
```

```
        let fileName = '';

        if ($files && $files.length > 1)
            fileName = ($this.attr('data-multiple-target') ||
            '').replace('{count}', $files.length);
        else if (e.target.value)
            fileName = e.target.value.split('\\').pop();

        if (fileName) {
            $label.find('.span-choose-file').html(fileName);
        } else {
            $label.html($label.html());
        }

    });
</script>
</body>
</html>
```

BAB III

SCREENSHOT INPUT-OUTPUT PROGRAM

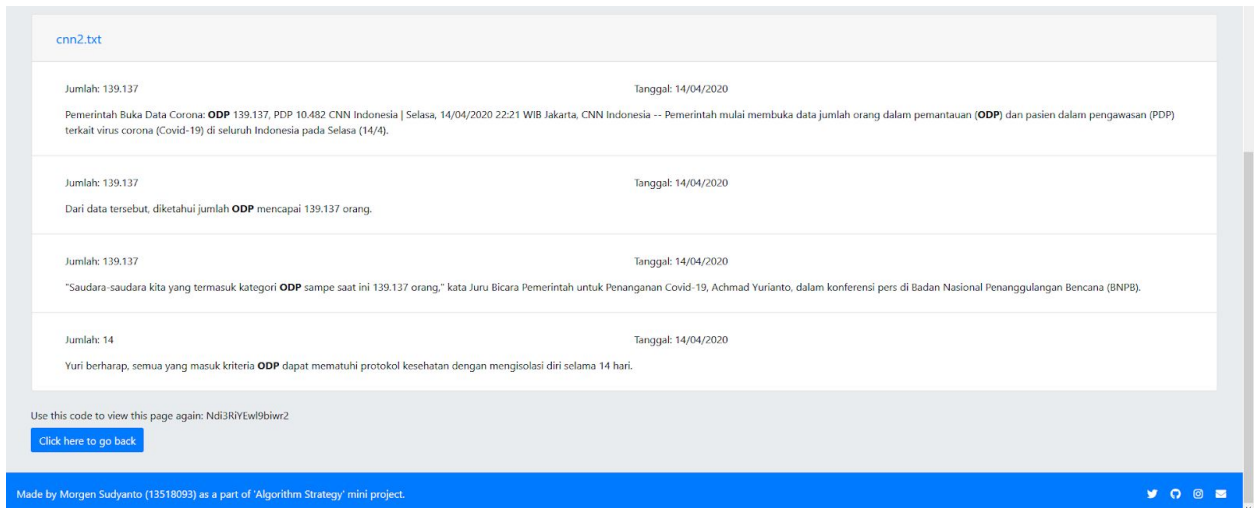
- I. Input dengan file
 - A. file: "cnn2.txt", "cnbc1.txt", "cnn1.txt"
 - B. keyword: ODP
 - C. algoritma: KMP

The screenshot shows the CoroFind web application interface. At the top, there is a blue header with the text "CoroFind". Below the header, the main content area has a light gray background. It starts with a "Welcome" message and a brief description: "This is a website that helps you extract information from news websites." Below this, there is a section titled "Start by clicking on one of these boxes below:" which contains four buttons: "Search by pasting text", "Search by uploading file(s)", "Upload your file(s):", and "Open with code". The "Search by uploading file(s)" button is selected. Below this button, there is a file upload section with a text input field showing "3 files selected" and a "Browse" button. Below the file upload section, there is a "Keyword:" label and a text input field containing "ODP". Below the keyword field, there is a "Choose an algorithm:" label and three radio buttons: "KMP" (selected), "BM", and "Regex". Below the algorithm selection, there is a "Submit" button. At the bottom of the interface, there is a blue footer with the text "Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project." and social media icons.

Gambar 3.1 Tampilan input dengan file

The screenshot shows the CoroFind web application interface displaying the results of a search. At the top, there is a blue header with the text "CoroFind". Below the header, the main content area has a light gray background. It starts with the text "You searched for ODP." and "Algorithm used: KMP". Below this, there are three sections, each representing a file: "cnbc1.txt", "cnn1.txt", and "cnn2.txt". Each section contains a "Jumlah:" (Count) and a "Tanggal:" (Date). The "cnbc1.txt" section shows a count of 261 and a date of 14 Mar. The "cnn1.txt" section shows a count of 20 and a date of 20 April. The "cnn2.txt" section is partially visible at the bottom. Each section also contains a brief description of the news article. At the bottom of the interface, there is a blue footer with the text "Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project." and social media icons.

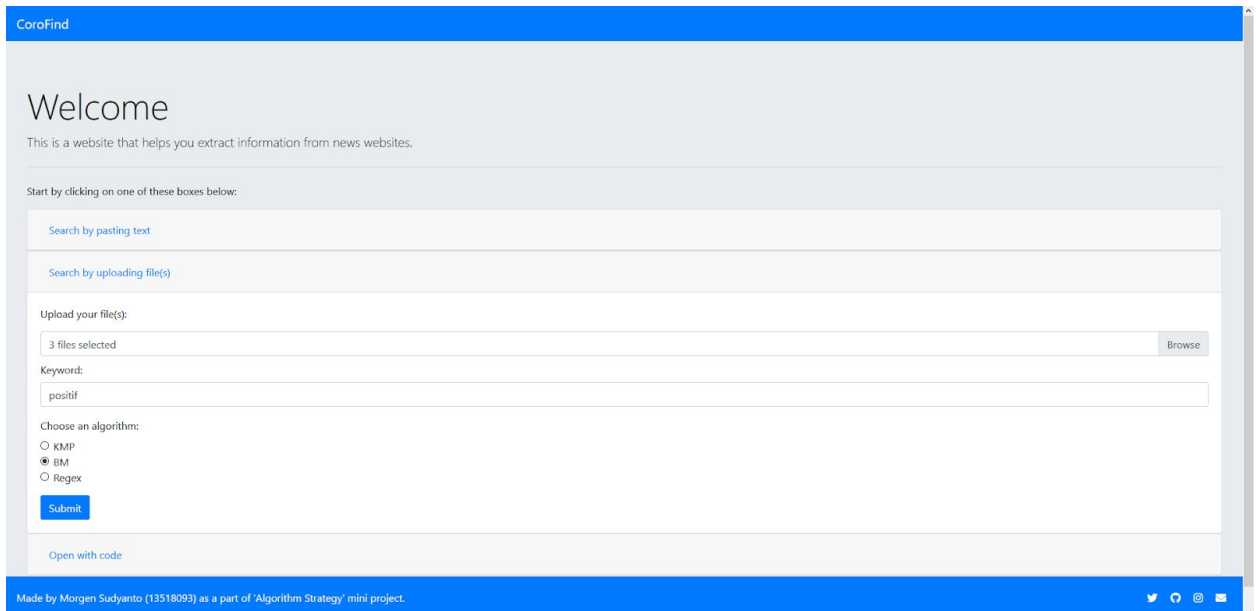
Gambar 3.2 Tampilan output 1



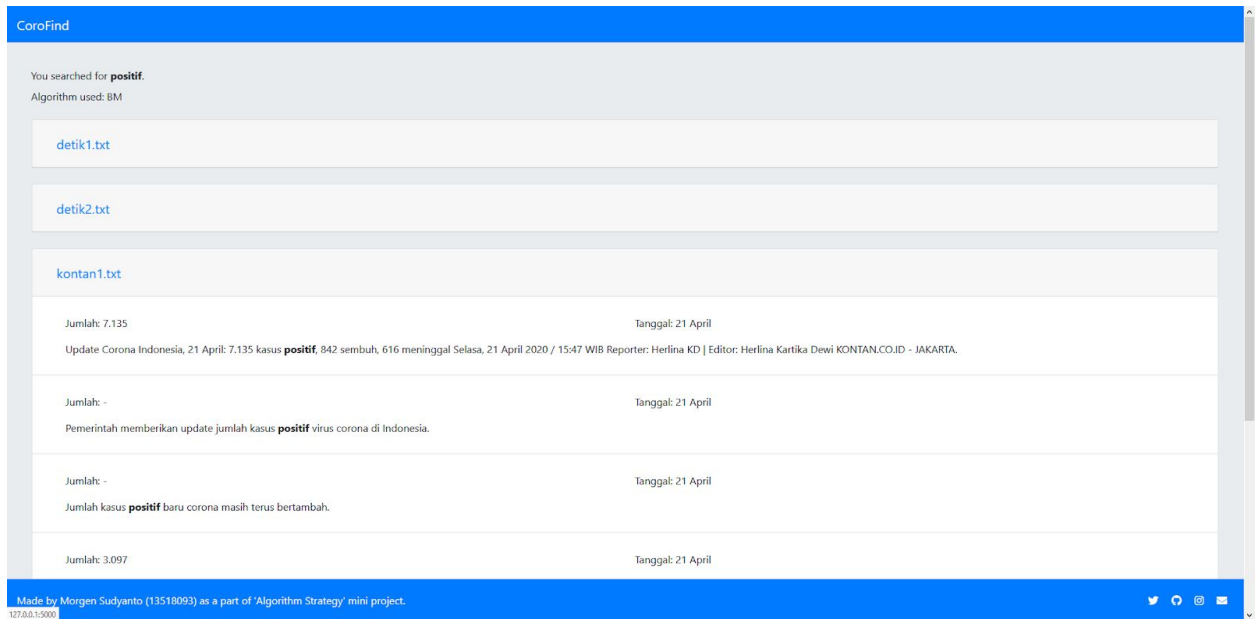
Gambar 3.3 Tampilan output 2

II. Input dengan file

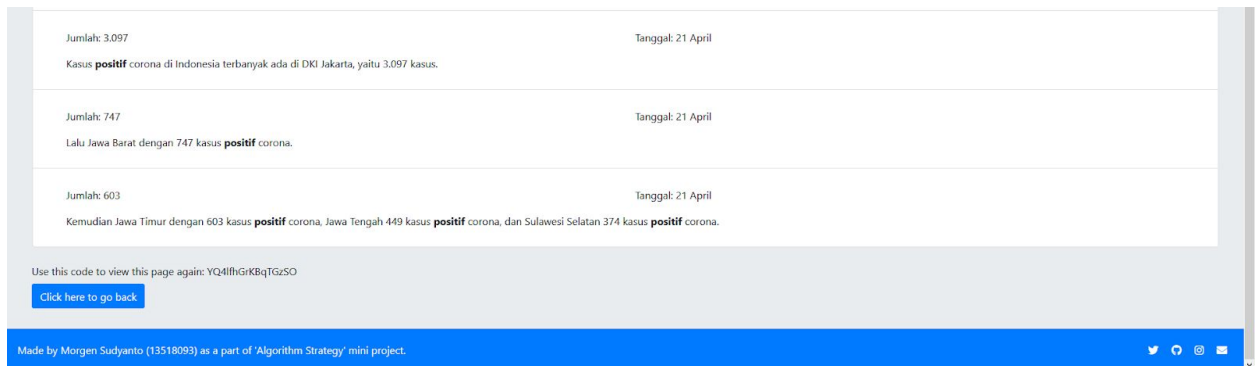
- A. file: "kontan1.txt" "detik1.txt" "detik2.txt"
- B. keyword: positif
- C. algoritma: BM



Gambar 3.4 Tampilan input dengan file



Gambar 3.5 Tampilan output 1



Gambar 3.6 Tampilan output 2

III. Input dengan text

- file: "kontan2.txt" (di copy-paste)
- keyword: positif
- algoritma: Regex

CoroFind

Welcome

This is a website that helps you extract information from news websites.

Start by clicking on one of these boxes below:

Search by pasting text

Input your text:

Kemudian Jawa Timur dengan 475 kasus positif corona, Banten 280 kasus positif corona, dan Jawa Tengah 278 kasus positif corona.
Sementara kasus meninggal terbanyak karena corona di Indonesia juga di DKI Jakarta, yaitu 241 orang, dan Jawa Barat ada 52 orang. Lalu Jawa Timur ada 41 kematian karena corona.

Keyword:

positif

Choose an algorithm:

☐ KMP
☐ BM
☒ Regex

Submit

Search by uploading file(s)

Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.7 Tampilan input dengan text

You searched for **positif**.
Algorithm used: Regex

Text	Tanggal: 14 April
Jumlah: 4.839 UPDATE Corona di Indonesia: 4.839 kasus positif , 426 sembuh, 459 meninggal Selasa, 14 April 2020 / 16:10 WIB Reporter: Herlina KD, Khomarul Hidayat Editor: Herlina Kartika Dewi KONTAN.CO.ID - JAKARTA.	Tanggal: 14 April
Jumlah: - Pemerintah memberikan update jumlah kasus positif virus corona di Indonesia.	Tanggal: 14 April
Jumlah: - Jumlah kasus positif baru corona masih terus bertambah.	Tanggal: 14 April
Jumlah: 2.335 Kasus positif corona di Indonesia terbanyak ada di DKI Jakarta, yaitu sebanyak 2.335 kasus.	Tanggal: 14 April
Jumlah: 530 Disusul Jawa Barat dengan 530 kasus positif corona.	Tanggal: 14 April
Jumlah: 475 Kemudian Jawa Timur dengan 475 kasus positif corona, Banten 280 kasus positif corona, dan Jawa Tengah 278 kasus positif corona.	Tanggal: 14 April

Use this code to view this page again: 2jcJpV1q25qfWZ5
[Click here to go back](#)

Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.8 Tampilan output

IV. Input dengan text

- A. file: "kumparan1.txt" (di copy-paste)
- B. keyword: PDP
- C. algoritma: KMP

CoroFind

Welcome

This is a website that helps you extract information from news websites.

Start by clicking on one of these boxes below:

Search by pasting text

Input your text:

Yuri mengungkapkan, per hari ini pukul 12.00 WIB, jumlah pasien positif virus corona naik 407 orang. Sehingga total kini ada sebanyak 5.923 pasien positif corona. Dari jumlah itu pasien positif COVID-19 yang meninggal bertambah 24 orang, total menjadi 520 orang. Sementara jumlah pasien virus corona yang sembuh bertambah 59 orang, menjadi 607 orang. "Kita bersyukur yang sembuh 607, dan akan terus meningkat dalam jumlah cukup besar," ucapnya.

Keyword:

PDP

Choose an algorithm:

☒ KMP
☐ BM
☐ Regex

Submit

Search by uploading file(s)

Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.9 Tampilan input dengan text

CoroFind

You searched for **PDP**.

Algorithm used: KMP

Text
<p>Jumlah: 12.610</p> <p>Tanggal: 17 April</p> <p>News 17 April 2020 16:14 Update 17 April ODP Corona di Indonesia 173.000, PDP 12.610 Pemerintah memperbarui jumlah orang dalam pemantauan (ODP) dan pasien dalam pengawasan (PDP) terkait virus corona di Indonesia.</p>
<p>Jumlah: 12.610</p> <p>Tanggal: 17 April</p> <p>Sementara itu, jumlah PDP di seluruh wilayah Tanah Air tercatat ada sebanyak 12.610 orang.</p>
<p>Jumlah: -</p> <p>Tanggal: 17 April</p> <p>Yuri menyebut PDP tersebut akan menjadi prioritas pemeriksaan corona menggunakan metode Polymerase Chain Reaction (PCR).</p>
<p>Jumlah: 12.610</p> <p>Tanggal: 17 April</p> <p>"PDP 12.610 yang nantinya akan jadi prioritas untuk diperiksa PCR real time," ucapnya.</p>

Use this code to view this page again: KobRYhnaTo2Treq

[Click here to go back](#)

Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.10 Tampilan output

V. Input dengan file

- file: "suara1.txt" "tirto1.txt"
- keyword: ODP
- algoritma: BM

CoroFind

Welcome

This is a website that helps you extract information from news websites.

Start by clicking on one of these boxes below:

Search by pasting text

Search by uploading file(s)

Upload your file(s):

2 files selected Browse

Keyword:

ODP

Choose an algorithm:

☐ KMP
☒ BM
☐ Regex

Submit

[Open with code](#)

Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.11 Tampilan input dengan file

CoroFind

You searched for **ODP**.
Algorithm used: BM

File	Results
suara1.txt	<p>Jumlah: - Tanggal: 13 April</p> <p>Jokowi Umumkan Data PDP dan ODP Virus Corona Pebransyah Ariefana Ummi Hadyah Saleh Senin, 13 April 2020 11:09 WIB Suara.com - Presiden Joko Widodo atau meminta data Pasien Dalam Pengawasan (PDP) dan Orang Dalam Pemantauan (ODP) virus corona diumumkan ke publik.</p> <p>Jumlah: - Tanggal: 13 April</p> <p>"Sehingga semua informasi ada baik mengenai jumlah Pasien Dalam Pengawasan (PDP), Orang Dalam Pemantauan (ODP) di daerah, jumlah yang meninggal, jumlah yang sembuh semua menjadi jelas dan terdata dengan baik," kata Presiden Joko Widodo di Istana Merdeka, Jakarta Senin (13/4/2020).</p> <p>Jumlah: - Tanggal: 13 April</p> <p>Sekali lagi data terpadu ini menyangkut PDP, ODP, positif, kemudian yang sembuh, yang meninggal, jumlah yang di-PCR berapa ada semua dan terbuka semua sehingga semua orang bisa mengakses data ini dengan baik," tegas Presiden.</p>
tirto1.txt	<p>Jumlah: 181.770 Tanggal: 21 April</p> <p>Adapun mereka yang ditetapkan menjadi Orang Dalam Pemantauan (ODP) sebanyak 181.770 jiwa.</p>

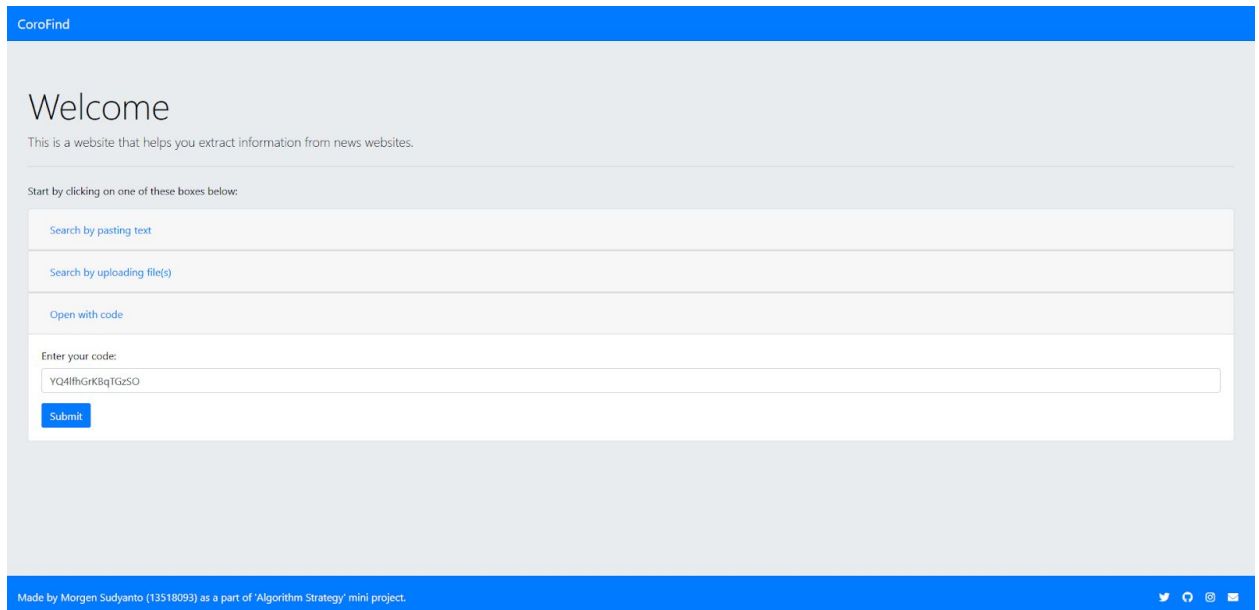
Use this code to view this page again: 8OLKB10GFu7FLSgi

Click here to go back

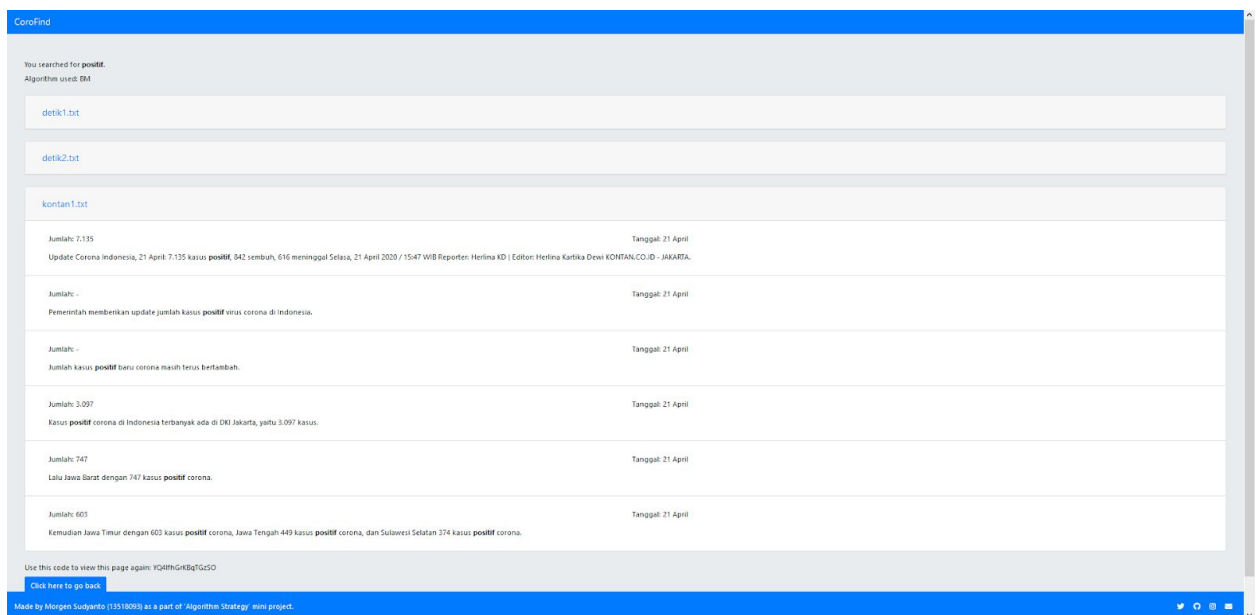
Made by Morgen Sudyanto (13518093) as a part of 'Algorithm Strategy' mini project.

Gambar 3.12 Tampilan output

VI. Input dengan menggunakan kode yang didapat pada tes II (YQ4lfhGrKBqTGzSO)



Gambar 3.13 Tampilan input dengan kode



Gambar 3.14 Tampilan output. Hasil ini sama seperti Gambar 3.5 dan 3.6

BAB IV

CHECKLIST KEBERHASILAN PROGRAM

Poin	Ya	Tidak
1. Program berhasil dikompilasi	V	
2. Program berhasil running	V	
3. Program dapat menerima input dan menuliskan output.	V	
4. Luaran sudah benar untuk data uji	V	

BAB V

DEPLOYMENT

Aplikasi ini dideploy di <https://corofind.herokuapp.com/>.

SUMBER

Wibisono, Yudi dan Khodra, Masayu Leylia (2020). Modul Praktikum Kuliah Pengantar Regular Expression. Diakses pada 21 April 2020.