

**IF2211 Strategi Algoritma
(Semester II Tahun Ajaran 2025/2026)**

Informasi ringkas perkuliahan

<i>Bobot SKS</i>	: 3
<i>Tims dosen</i>	: K1 - Prof. Dr. Ir. Rinaldi, M.T – Koordinator kuliah K2 – Dr. Eng. Ir. Rila Mandala, M.Eng K3 - Dr. Techn. M. Catur Zuhri Chandra, S.T, M.T/ Dr. Fazat Nur Azizah, S.T, M.Sc./ Tricya Esterina Widagdo, S.T, M.Sc
E-mail	: rinaldi@staff.stei.itb.ac.id , rinaldi@iitb.ac.id rila@staff.stei.itb.ac.id , rila@itb.ac.id catur@staff.stei.itb.ac.id , catur@itb.ac.id fazat@staff.stei.itb.ac.id , fazat@itb.ac.id cia@staff.stei.itb.ac.id , tricia@itb.ac.id
URL kuliah	: 1. Edunex (https://edunex.itb.ac.id/) 2. http://informatika.stei.itb.ac.id/~rinaldi.munir
<i>Asisten</i>	: 2 orang per kelas

Tujuan Umum Kuliah:

Mahasiswa memahami beberapa strategi yang digunakan dalam memecahkan persoalan secara sangkil sesuai dengan karakteristik persoalan tersebut.

Tujuan Khusus:

Mahasiswa mampu:

1. Memilih strategi yang tepat untuk menyelesaikan suatu persoalan.
2. Membuat program/aplikasi untuk menyelesaikan persoalan dengan strategi yang tepat.

Capaian Pembelajaran Mata Kuliah (CPMK):

1. CPMK1. Menyusun solusi masalah komputasi dengan berbagai strategi algoritma, bergantung pada karakteristik persoalan, kemudian mendesain algoritma penyelesaian masalah
2. CPMK2. Mengimplementasikan desain algoritma menjadi sebuah program aplikasi
3. CPMK3. Menulis makalah ilmiah yang berisi hasil studi dan implementasi penyelesaian masalah dengan menggunakan berbagai strategi algoritma.

4. CPMK4. Menulis laporan penyelesaian masalah komputasi yang diberikan di dalam tugas
5. CPMK5. Mendemokan aplikasi untuk mendapatkan feedback

Prasyarat Kuliah:

1. IF1220 Matematika Diskrit
2. IF2110 Algoritma dan Pemrograman 2

Pokok Bahasan:

1. Pengantar Strategi Algoritma
3. Algoritma *brute force*
4. Algoritma *greedy*
5. Algoritma *divide and conquer*
6. Algoritma *decrease and conquer*
6. DFS dan BFS
7. Algoritma *backtracking*
8. Algoritma *branch and bound*
9. Algoritma *A**, *Best First Search*, dan UCS
10. Program dinamis (*dynamic programming*)
11. *String matching + regular expression (Regex)*
12. Teori P, NP, dan NP-Completes

Referensi kuliah:

1. Anany Levitin, *Introduction to the Design & Analysis of Algorithms*, Addison-Wesley, 2003.
2. Bhardwaj, Anuj; Verma, Parag, *Design and Analysis of Algorithm*, Alpha Science International Ltd., 2017, can be accessed at: <http://portal.igpublish.com/iglibrary/obj/APSB0000252?dtbs=&searchid=1573133164973plapRvm7MQJMAwkcwThUs> from <https://lib.itb.ac.id/>
3. Khan Academy, *Computer Science: Algorithm*, can be accessed at <https://www.khanacademy.org/computing/computer-science/algorithms>
4. Coursera, *Data Structures and Algorithms Specialization*, 2019, can be accessed at: <https://www.coursera.org/learn/algorithmic-toolbox?specialization=data-structures-algorithms>
5. Stuart J Russell & Peter Norvig, *Resources of topics in Artificial Intelligence: A Modern Approach*, 3rd Edition, Global Edition Paperback, Pearson, 2016, <http://aima.cs.berkeley.edu/>
6. Brandons Kerritt, *All You Need to Know About Big O Notation [Python Examples]*, 2019, can be accessed at <https://dev.to/brandonskerritt/all-you-need-to-know-about-big-o-notation-python-examples-2k4o>
7. Coursera, *Machine Learning Clustering and Retrieval: Complexity of Brute Force Algorithm*, 2019, can be accessed at <https://www.coursera.org/lecture/ml-clustering-and-retrieval/complexity-of-brute-force-search-5R6q3>

8. Rinaldi Munir, Diktat kuliah IF2251 Strategi Algoritmik, Teknik Informatika ITB
9. Marin Vlastelica Pogančić, The Branch and Bound Algorithm, 2019, can be accessed at <https://towardsdatascience.com/the-branch-and-bound-algorithm-a7ae4d227a69>
10. Jurafsky and James H. Martin, *Speech and Language Processing*, 2019, can be accessed at <https://web.stanford.edu/~jurafsky/slp3/>
11. Chua Hock-Chuan, Regular Expressions (Regex), 2018, can be accessed at <https://www.ntu.edu.sg/home/ehchua/programming/howto/Regexe.html>
12. Ellis Horowitz & Sartaj Sahni, *Computer Algorithms*, Computer Science Press, 1998.
13. Richard E. Neapolitan, *Foundations of Algorithms*, D.C. Heath and Company, 1996
14. Thomas H. Cormen, *Introduction to Algorithms*, The MIT Press, 1992.

Penilaian :

1. Ujian Tengah Semester (UTS) – 1 kali
2. Ujian Akhir Semester (UAS) – 1 kali
3. Tugas Besar (Tubes) – 3 buah, per kelompok
4. Tugas Kecil (Tucil) – 3 buah, per orang/2 orang
4. Makalah (di akhir semester) – 1 buah
5. Kehadiran

Tugas pemrograman adalah tugas besar berkelompok. Program dikembangkan dengan menggunakan kakas pemrograman berbasis GUI (*Graphical User Interface*). Tugas pertama direncanakan menggunakan kakas pemrograman dengan GUI menggunakan bahasa C++ (misalnya Visual C++). Tugas kedua menggunakan C# dengan kakas Visual Studio atau yang sejenis, dan tugas ketiga menggunakan Java (NetBeans, Eclipse) + PHP. Tugas Kecil adalah tugas perorangan atau per-dua orangan dengan bahasa pemrograman yang ditentukan (C, C++, Java, Python, Golang, Rust, dll).

Perkiraan batas-batas nilai untuk perhitungan indeks (tidak tetap, bisa berubah)

Nilai Akhir ≥ 81 , indeks = A
 $76 \leq$ Nilai Akhir < 81 , indeks = AB
 $71 \leq$ Nilai Akhir < 76 , indeks = B
 $65 \leq$ Nilai Akhir < 71 , indeks = BC
 $55 \leq$ Nilai Akhir < 65 , indeks = C
 $45 \leq$ Nilai Akhir < 55 , indeks = D
 Nilai Akhir < 45 , indeks = E