

**Tugas Kecil 1**  
**IF2211 Strategi Algoritma**  
**Penyelesaian Permainan Kartu 24 dengan Algoritma Brute Force**  
**SEMESTER 2 TAHUN 2022/2023**



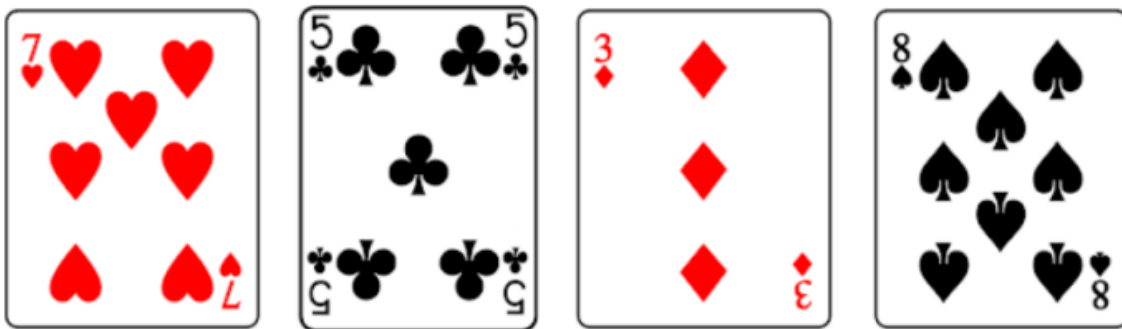
Disusun oleh :  
Dewana Gustavus Haraka Otang      13521173

TEKNIK INFORMATIKA  
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG

2023

## Bab I Deskripsi Masalah

Permainan kartu 24 adalah permainan kartu aritmatika dengan tujuan mencari cara untuk mengubah 4 buah angka random sehingga mendapatkan hasil akhir sejumlah 24. Permainan ini menarik cukup banyak peminat dikarenakan dapat meningkatkan kemampuan berhitung serta mengasah otak agar dapat berpikir dengan cepat dan akurat. Permainan Kartu 24 biasa dimainkan dengan menggunakan kartu remi. Kartu remi terdiri dari 52 kartu yang terbagi menjadi empat suit (sekop, hati, keriting, dan wajik) yang masing-masing terdiri dari 13 kartu (As, 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, dan King). Yang perlu diperhatikan hanyalah nilai kartu yang didapat (As, 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, dan King). As bernilai 1, Jack bernilai 11, Queen bernilai 12, King bernilai 13, sedangkan kartu bilangan memiliki nilai dari bilangan itu sendiri. Pada awal permainan moderator atau salah satu pemain mengambil 4 kartu dari dek yang sudah dikocok secara random. Permainan berakhir ketika pemain berhasil menemukan solusi untuk membuat kumpulan nilainya menjadi 24. Pengubahan nilai tersebut dapat dilakukan menggunakan operasi dasar matematika penjumlahan (+), pengurangan (-), perkalian ( $\times$ ), divisi (/) dan tanda kurung ( ). Tiap kartu harus digunakan tepat sekali dan urutan penggunaannya bebas. (Paragraf di atas dikutip dari sini: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2015-2016/Makalah2016/MakalahStima-2016-038.pdf>).



MAKE IT 24

## Bab II Teori Dasar

Salah satu strategi algoritma yang dapat digunakan untuk menyelesaikan permasalahan diatas adalah dengan menggunakan algoritma bruteforce. Prosedur algoritma bruteforce yang digunakan untuk menyelesaikan masalah adalah sebagai berikut:

1. Urutkan kartu sebelum mencari seluruh permutasi kartu
2. Simpan seluruh permutasi kartu yang mungkin
3. Cari dan simpan seluruh kombinasi operasi yang mungkin
4. Iterasi seluruh permutasi kartu
5. Untuk setiap permutasi kartu, Iterasi kombinasi operasi
6. Untuk setiap kombinasi operasi, Iterasi Seluruh kemungkinan urutan melakukan operasi

7. Simpan solusi pada sebuah list yang berisi string yang menyatakan bagaimana susunan kartu serta operasi dan urutan operasi yang digunakan sehingga dapat menghasilkan angka 24.

Berdasarkan penjabaran operasi diatas didapat kompleksitas waktu program adalah  $O(n \log(n) + n! + k^{n-1} + n! k^{n-1} (n-1)!)$ , dengan n adalah banyak kartu yang dipakai dan k adalah jumlah operasi yang dapat dilakukan. Kompleksitas waktu masih bisa disederhanakan menjadi  $O((n!)^2 k^n)$ .

### Bab III Implementasi

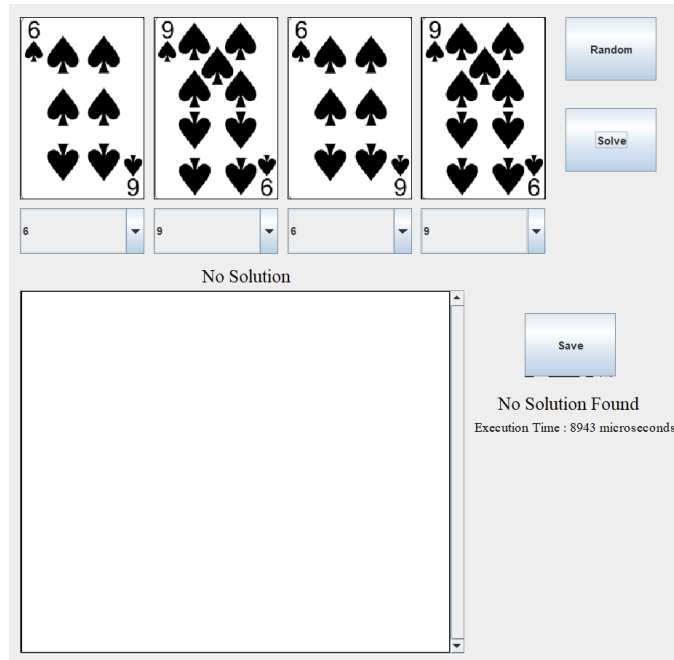
Pada implementasi digunakan struktur data pecahan pada operasi aritmatika sehingga mencegah adanya eror pada presisi desimal. Program yang menangani struktur data pecahan terdapat pada folder src dengan nama file Fraction.java. Operasi dasar pecahan yang dibuat menangani operasi dasar yang diperlukan pada permainan 24 yaitu +, -, ×, ÷.

Implementasi algoritma brute force yang digunakan berada pada folder src dengan nama file Solver.java. Untuk menyelesaikan langkah pertama digunakan algoritma counting sort yang membuat array frekuensi lalu membuat kembali array dengan keadaan terurut dengan cara mengiterasi satu per satu nilai pada array frekuensi. Untuk menyelesaikan langkah kedua dan ketiga digunakan algoritma backtrack dengan rekursif. Untuk menyelesaikan langkah keenam seluruh urutan operasi diketik secara manual (hard coded) lalu dilakukan operasi sesuai urutan yang ditentukan. Untuk menyelesaikan langkah ketujuh program akan menyimpan hasil pada struktur data array dinamik yang dimiliki Java yaitu ArrayList, setelah mendapat seluruh solusi program akan memberikan solusi tersebut kepada program utama (GUI) lalu akan mereset kembali array solusi sehingga pada perhitungan selanjutnya array sudah siap digunakan.

Untuk implementasi User Interface pembuat memilih menggunakan GUI sehingga tidak perlu menangani input dan output yang diluar format karena pengguna akan dipaksa menggunakan program hanya dengan tombol-tombol yang disediakan. Implementasi GUI berada pada folder src dengan nama file GUI.Java. Implementasi GUI dijadikan sebagai program utama sehingga apabila ingin menjalankan program dapat dilakukan dengan menjalankan file GUI.Java. Implementasi GUI memanfaatkan library Swing yang dimiliki Java serta beberapa library yang membantu membaca gambar dan menyimpan file solusi.

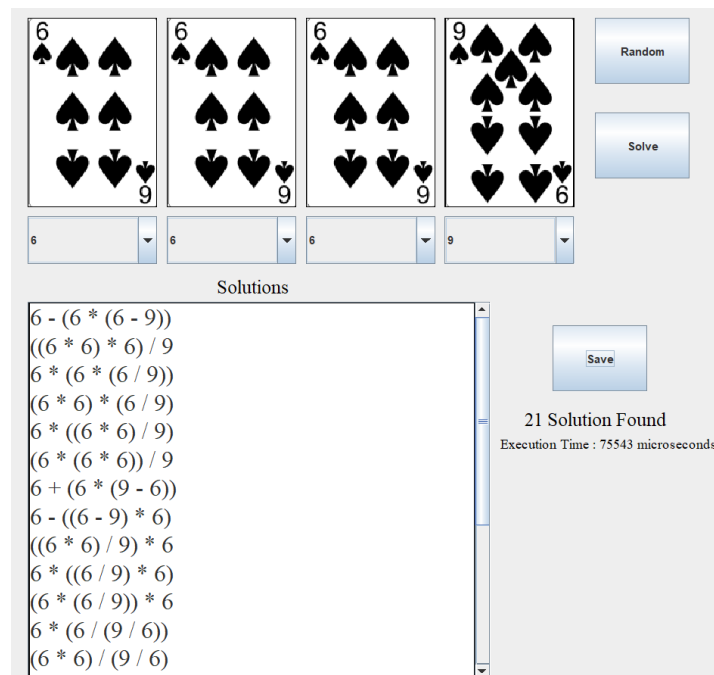
### Bab IV Eksperimen

1. Test 1  
Contoh masukan yang tidak memiliki solusi.



## 2. Test 2

Contoh masukan dengan solusi banyak, program membutuhkan waktu eksekusi yang lama.



## 3. Test 3

Contoh masukan dengan solusi banyak, program membutuhkan waktu eksekusi yang lama.

6 9 4 2

6 9 4 2

Solutions

```

(2 + (4 / 6)) * 9
2 * (4 * (9 - 6))
(2 * 4) * (9 - 6)
2 * ((9 - 6) * 4)
(2 * (9 - 6)) * 4
((4 - 2) * 9) + 6
4 * (2 * (9 - 6))
(4 * 2) * (9 - 6)
((4 / 2) * 9) + 6
(4 / (2 / 9)) + 6
((4 / 6) + 2) * 9
(4 * 9) - (6 * 2)
4 * ((9 - 6) * 2)

```

Save

45 Solution Found  
Execution Time : 24895 microseconds

#### 4. Test 4

Contoh masukan dengan solusi banyak, program membutuhkan waktu eksekusi yang lama.

A 2 3 4

A 2 3 4

Solutions

```

((A + 2) + 3) * 4
(A + (2 + 3)) * 4
((A * 2) * 3) * 4
A * (2 * (3 * 4))
(A * 2) * (3 * 4)
A * ((2 * 3) * 4)
(A * (2 * 3)) * 4
((A * 2) * 4) * 3
A * (2 * (4 * 3))
(A * 2) * (4 * 3)
A * ((2 * 4) * 3)
(A * (2 * 4)) * 3
((A + 3) + 2) * 4

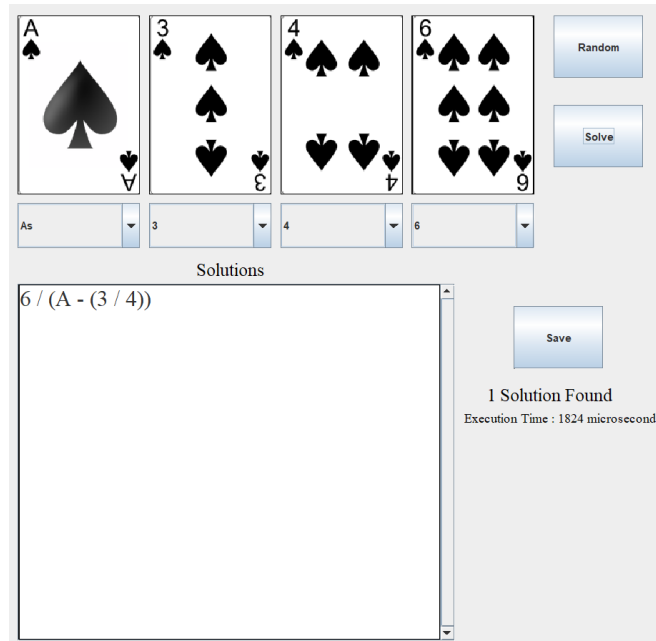
```

Save

242 Solution Found  
Execution Time : 51504 microseconds

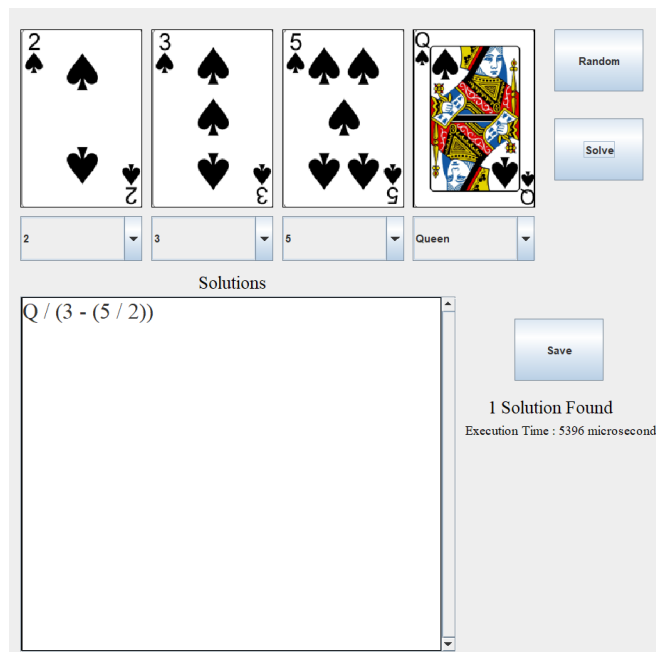
#### 5. Test 5

Masukan dari <https://samidavies.wordpress.com/2017/04/14/the-hardest-games-of-24/>, program yang memiliki solusi sedikit memiliki waktu eksekusi sebentar.



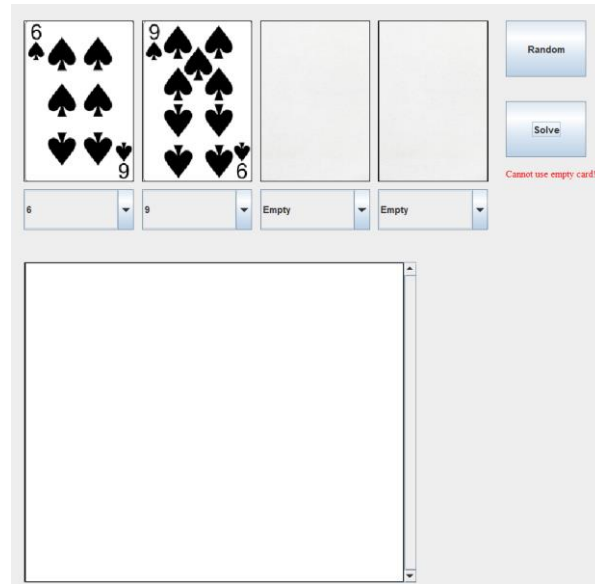
6. Test 6

Masukan dari <https://samidavies.wordpress.com/2017/04/14/the-hardest-games-of-24/>, program yang memiliki solusi sedikit memiliki waktu eksekusi sebentar.



7. Test 7

Contoh masukan dengan kartu kosong, algoritma penyelesai tidak bisa dijalankan dan masukan tidak dapat disimpan.



## Bab V Kesimpulan, Saran, dan Refleksi

Meskipun algoritma brute force yang dibuat memiliki kompleksitas waktu  $O((n!)^2 k^n)$  yang merupakan kompleksitas yang cukup besar, algoritma tetap dapat berjalan dalam waktu yang singkat dikarenakan nilai  $n$  dan  $k$  yang kecil. Dalam segi UI program masih dapat ditingkatkan pada bagian tampilan kartu yang hanya menggunakan 1 setelan kartu dan dapat dikembangkan sehingga tidak ada 2 buah kartu dengan angka yang sama dan setelan kartu yang sama. Pada kombinasi operasi “a op1 (b op2 (c op3 d))” menimbulkan banyak solusi yang mirip dengan “((a op1 b) op2 c) op3 d” tetapi tetap perlu dimasukkan karena jika tidak akan ada masukan yang tidak memiliki hasil seperti masukan “A 3 4 6”, karena hal tersebut pada luaran solusi terdapat banyak solusi duplikat yang mirip dan belum terpikir cara penanganan yang lebih efektif.

## Bab VI Referensi

[www.youtube.com/watch?v=Kmg00avvEw](http://www.youtube.com/watch?v=Kmg00avvEw)

<https://www.javatpoint.com/java-swing>

<https://stackoverflow.com/questions/6714045/how-to-resize-jlabel-imageicon>

<https://stackoverflow.com/questions/15173325/execute-an-action-when-an-item-on-the-combobox-is-selected>

<https://www.geeksforgeeks.org/write-a-c-program-to-print-all-permutations-of-a-given-string/>

<https://www.geeksforgeeks.org/distinct-permutations-string-set-2/>

<https://stackoverflow.com/questions/30222157/displaying-arrayliststring-in-jtextarea>

<https://www.geeksforgeeks.org/find-time-taken-to-execute-the-tasks-in-a-based-on-the-order-of-execution-in-b/>

<https://stackoverflow.com/questions/24113453/how-do-i-code-the-function-of-the-save-button-in-java>

<https://samidavies.wordpress.com/2017/04/14/the-hardest-games-of-24/>

<https://possiblywrong.wordpress.com/2017/03/11/the-hardest-24-puzzles/>

<https://liao-frank.github.io/24-Math-Game-Solver/>

<https://stackoverflow.com/questions/291115/java-swing-using-jscrollpane-and-having-it-scroll-back-to-top>

<https://stackoverflow.com/questions/56104371/fix-unchecked-cast-from-object-to-comboboxstring-javafx>

<https://stackoverflow.com/questions/26340990/problems-with-images-when-compiled>

<https://stackoverflow.com/questions/8055501/how-to-copy-in-bash-all-directory-and-files-recursive>

<https://www.technolox.com/tutorials/java-to-exe/>

<https://stackoverflow.com/questions/18288141/how-to-turn-folder-into-jar-code>

<https://stackoverflow.com/questions/22682809/how-can-i-change-manifest-file-in-java>

## Bab VII Lampiran

Link repository github : [https://github.com/DewanaGustavus/Tucil1\\_13521173](https://github.com/DewanaGustavus/Tucil1_13521173)

### Checklist program

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan	✓	
2. Program berhasil running	✓	
3. Program dapat membaca input / generate sendiri dan memberikan luaran	✓	
4. Solusi yang diberikan program memenuhi (berhasil mencapai 24)	✓	
5. Program dapat menyimpan solusi dalam file teks	✓	