Laporan Dokumentasi Tugas Solver
Desain Analisis & Algoritma

Disusun oleh :

1. **M. Maulana Gian Pranaja (L0123077)**
2. **Primus Putra Prima (L0123111)**

Dosen Pengampu :
Fajar Muslim S.T., M.T

Program Studi Informatika
Fakultas Teknologi Informasi dan Sains Data
Universitas Sebelas Maret

2024

PENJABARAN PROBLEM

20 4-Solver berguna untuk menyelesaikan sebuah problem. Problem tersebut adalah bagaimana saja cara untuk menyusun 4 angka dan 5 operator matematika (+, -, *, /, dan '**()**') supaya mendapatkan hasil akhir sebanyak **20**.

Problem ini juga menimbulkan 2 kemungkinan dalam bagaimana problem tersebut diselesaikan. Karena sifat 4 angka yang dapat dimasukkan sesuai kehendak pengguna, maka 4 angka tersebut memiliki kemungkinan untuk dapat berkombinasi untuk menghasilkan hasil 20, atau sama sekali tidak memungkinkan.

Ketika program menemukan satu atau lebih kombinasi dari keempat angka, maka program akan memberikan output berupa solusi-solusi tersebut. Namun jika tidak ditemukan satupun kombinasi dari 4 angka yang dimasukkan yang dapat menghasilkan 20, maka program akan memberikan output bahwa tidak dapat ditemukan penyelesaian.

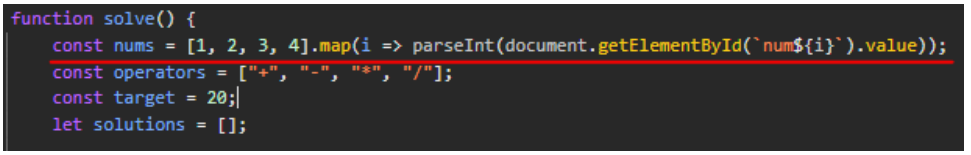
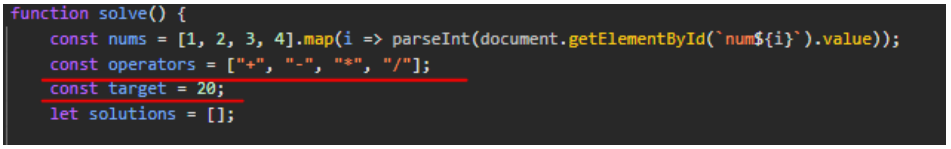
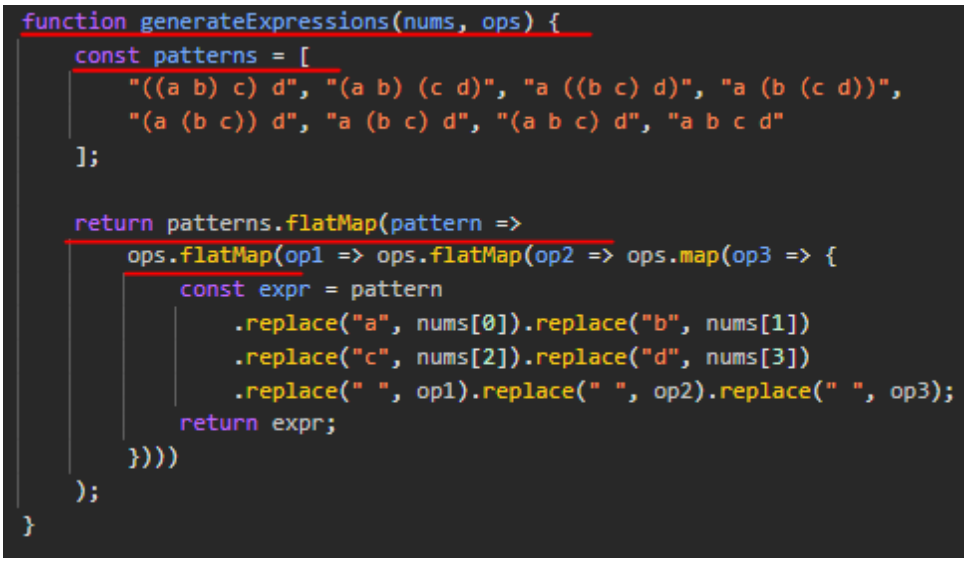
DEMONSTRASI PROGRAM (DEMO)

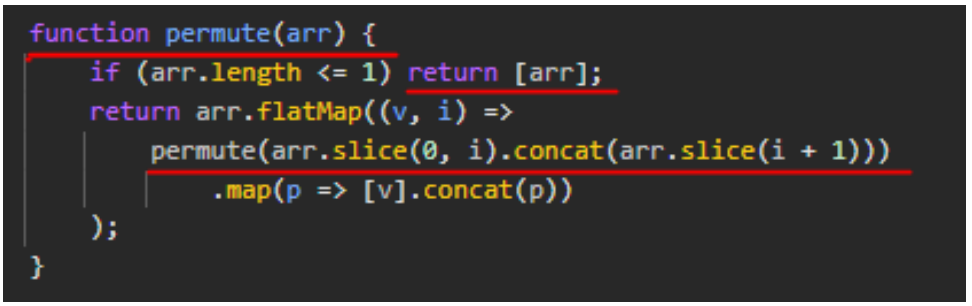
Tutorial 20 4 solver web

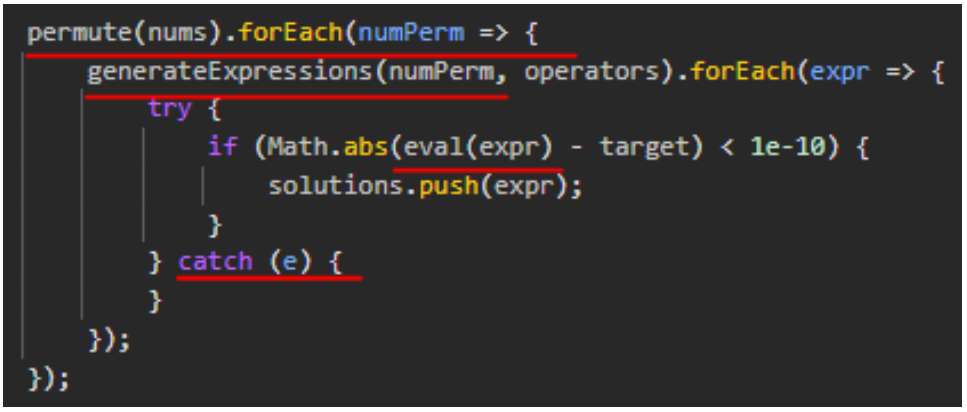
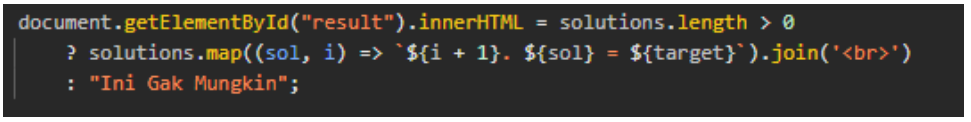
1. masuk ke link https://kingshellter.github.io/4solver20-Tugas-DAA/
2. masukan angka 2,3,5,10
3. klik solve, akan muncul semua solusi yang menghasilkan 20

:/ > video ada di repositories github

Penjelasan Implementasi

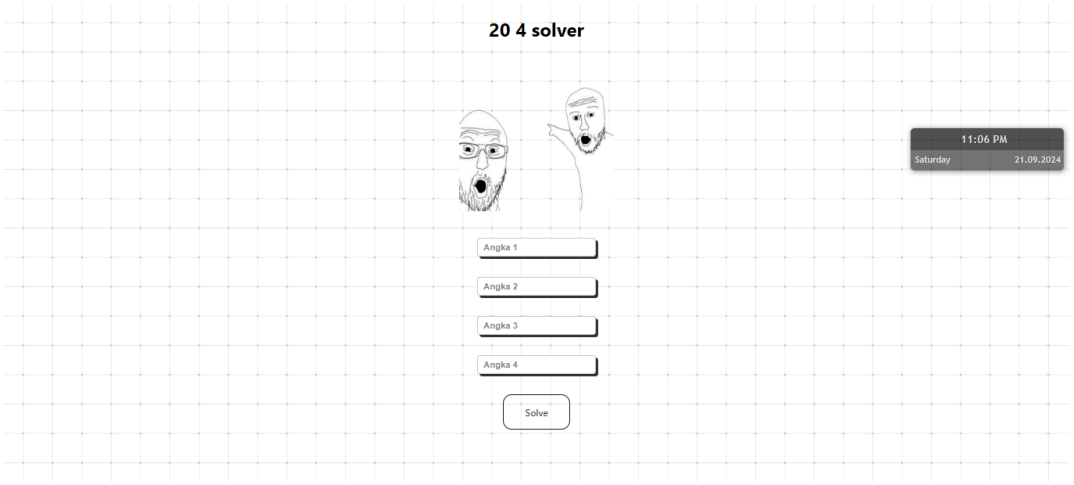
No.	Implementasi	PERAN
1.	Input Collection	 Line ini berguna untuk menciptakan sebuah *array* yang terdiri atas 4 angka. Baris ini menggunakan 'map' untuk mengiterasikan [1, 2, 3, 4], dan mengambil nilai dari input HTML dengan ID num1, num2, num3, dan num4, dan kemudian dikonversikan ke dalam bentuk integer.
2.	Penggunaan Constant	 Bagian ini adalah yang menentukan semua operator matematika, serta yang menentukan hasil akhir yang ingin ditemukan. Pada kasus ini, hasil akhir di-set 20. Namun nilai ini bisa diubah-ubah sesuai kehendak programmer.
3.	Expression Generator	 Bagian ini sangat penting dalam pengoperasian kode. Bagian ini menentukan seluruh format kombinasi angka dan operator matematika yang mungkin dalam menemukan hasil

		1. Baris **‘patterns**’ menentukan seluruh kemungkinan diletakkannya *parentheses* (kurung buka-tutup) pada operator matematika 2. Kemudian dilanjutkan dengan panggilan ‘**flatMap**’ yang berfungsi untuk memberikan seluruh kombinasi operator matematika pada setiap ‘**patterns**’. Maka, ‘**flatMap**’ nantinya akan memberikan seluruh kombinasi operator matematika pada setiap pola *parentheses* 3. Untuk setiap kombinasi, ‘a b c d’ akan digantikan dengan angka yang dimasukkan pengguna
4.	Fungsi Permutasi	 Bagian ini merupakan fungsi rekursif yang akan membuat seluruh kemungkinan permutasi yang bisa dioperasikan pada *array* bilangan tersebut. Cara kerjanya 1. Ketika *array* memiliki 1 elemen atau kurang, maka akan dikembalikan ke *array* itu sendiri 2. Ketika syarat terpenuhi, maka a. Elemen tersebut akan dihapus, dan elemen sisanya akan dipermutasikan b. Kemudian setelah selesai dipermutasikan, elemen yang dihapus tersebut akan ditambahkan kembali pada setiap permutasi yang sudah terjadi

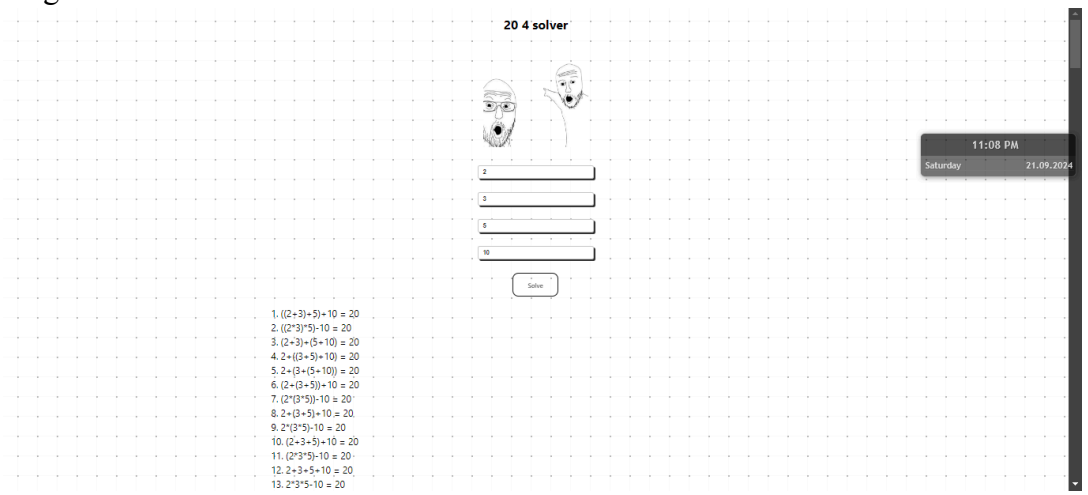
5.	Algoritma Utama	 Bagian ini menggabungkan seluruh fungsi sebelumnya 1. Membuat seluruh permutasi pada setiap angka yang dimasukkan oleh pengguna 2. Untuk setiap permutasi, maka semua kemungkinan 'expression' (semua kemungkinan operasi matematika) akan dibentuk 3. Untuk setiap 'expression', akan dicek menggunakan '**eval()**' 4. Jika 'expression' mampu menghasilkan hasil 20, maka 'expression' akan ditambahkan pada array 'solutions' 5. Penggunaan 'try-catch' diimplementasikan guna menangkap galat, andai saja ada expression yang mengoperasikan pembagian dengan 0
6.	Output	 Ketika ada solusi yang ditemukan, maka bagian ini akan membentuk numbered list dengan menambahkan '=20' di akhir. Namun ketika tidak ada solusi yang ditemukan, maka akan menampilkan "Ini Gak Mungkin".

PENGUJIAN KODE

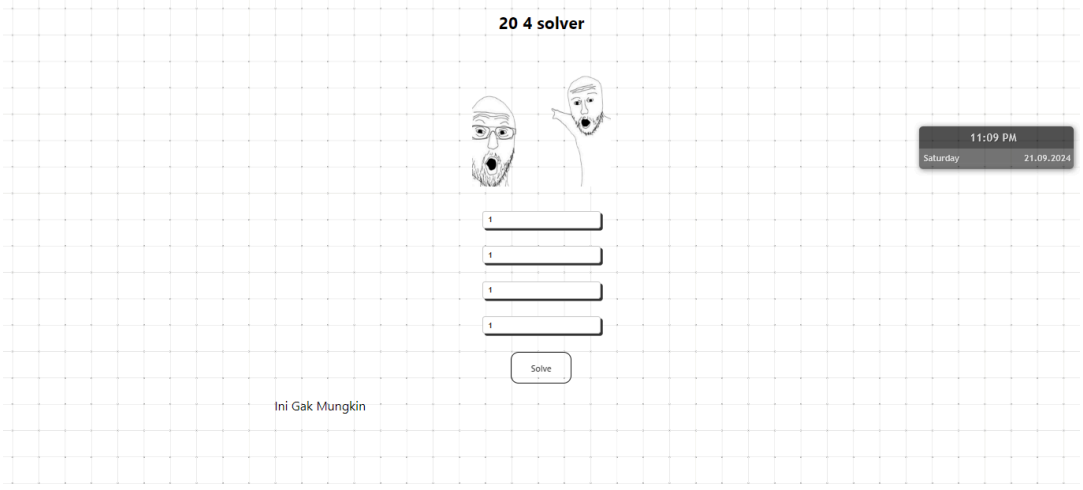
1. Tampilan UI



2. Program menemukan solusi



3. Ketika keempat angka tidak memenuhi



PERAN ANGGOTA KELOMPOK	
Anggota	Peran
M. Maulana Gian Pranaja	Desain UI, Web dev, troubleshooting code, video demo, github, JS Developer, laporan dokumentasi.
Primus Putra Prima	Javascript Developer, Implementasi, Penjabaran Problem, Pengujian Kode, laporan dokumentasi.