

TUGAS BESAR
MATA KULIAH IF 2180
SOSIO INFORMATIKA DAN PROFESIONALISME



Kelas 03

1. Ahmad Fauzan Putra (13524141)

INSTITUT TEKNOLOGI BANDUNG

1. Algoritma

- Algoritma ini berjalan dengan meletakkan sebuah ratu pada awal dari tiap baris
- Parse Input berdasarkan newline atau space untuk membaginya menjadi baris dan kolom
- Cek apakah tiap baris memiliki jumlah kolom yang sama, Cek apakah jumlah baris dan kolom sama
- Cek apakah jumlah warna sama dengan baris/kolom, cek apakah warna berkelompok dalam satu kelompok utama (Menggunakan Depth First Search, tidak dijelaskan pada laporan karena tidak jadi wajib).
- Program berjalan secara rekursif menggunakan fungsi loopN, menerima variabel d yang menandakan pada row berapa fungsi tersebut berjalan.
- Pada fungsi akan dilakukan for dimana jika bukan merupakan row paling akhir akan memasuki fungsi tersebut lagi namun pada baris dibawahnya ($d+1$), lalu melakukan increment pada posisi ratu pada baris d.
- Jika berada pada baris paling akhir, program akan memasuki fungsi confirmN setelah tiap increment posisi ratu, bisa dipastikan tiap posisi akan melalui pengecekan.
- Setiap kali sudah berlangsung pengecekan sebanyak k kali, program akan mengeluarkan output berupa bentuk dari papan pada saat pengecekan terakhir.
- Pengecekan dilakukan dengan pertama kali memasukan tiap posisi ratu kedalam sebuah set lalu cek apakah jumlah posisi unik sama dengan jumlah baris/kolom/warna

- Jika ya lanjut dengan melakukan pengecekan apakah sudah menempati semua warna dengan cara memasukan warna yang ditempati oleh tiap kedalam sebuah set character, lalu cek apakah jumlah karakter unik sama dengan jumlah warna.
- Jika ya, cek apakah posisi dari masing masing ratu tidak memiliki ratu lain pada diagonal atas kiri dan atas kanan. dengan melakukannya untuk setiap baris bisa dipastikan tidak akan ada ratu lain pada diagonal untuk hasil yang valid.
- Jika menemukan hasil yang valid, maka program akan mengubah boolean found menjadi true, dan menghentikan fungsi pencarian menggunakan break.
- Program akan output “Tidak ada solusi” jika variabel found adalah false.
- Jika true program akan output papan dengan posisi ratu yang tervalidasi benar bersama dengan jumlah kasus yang ditinjau , waktu pencarian (yang dimulai pada awal fungsi loop dan akhir fungsi loop).

2.Source Code

Karena Program terlalu panjang maka hanya akan dijelaskan secara singkat apa fungsi yang dilaksanakan oleh masing masing file.

<p>Program utama yang akan di compile, yang mencakup dua file lainnya yaitu normal.cpp dan Optimized.cpp. Program ini pada dasarnya hanyalah tampilan luar atau UI yang kemudian</p>	<p>gui.cpp</p>
--	-----------------------

memanggil fungsi dari normal.cpp atau Optimized.cpp. Program ini menggunakan multithreading agar UI tetap berjalan lancar meskipun sedang memproses papan permainan.	
Algoritma brute force naif di implementasi pada file ini.	normal.cpp
Algoritma yang memanfaatkan pruning cabang pencarian yang sudah pasti salah di implementasi pada file ini	Optimized.cpp

3. Tangkapan Layar

AAABBCCCD
ABBBBCECD
ABBBDCED
AAABDCCCD
BBBBDDDDD
FGGGDDHDD
FGIGDDHDD
FGIGDDHDD
FGGGDDHHH

Load txt

Process

Optimized Process

Save Output

Output Interval in Cases (Higher is lower)

100000

Output Interval in Milliseconds

3000.

A A A B B C C # D
A B B B # C E C D
A B B B D C # C D
A # A B D C C C D
B B B B D # D D D
F G G # D D H D D
G I G D D H D D
F G # G D D H D D
F G G G D D H H #

FOUND
Waktu Pencarian : 8.29513ms
Banyak Kasus Yang Di Tinjau: 27947

1.

AABB
AABB
CCDD
CCDD

Load txt

Process

Optimized Process

Save Output

Output Interval in Cases (Higher is lower)

Output Interval in Milliseconds

100000

3000

A # B B
A A B #
C D D
C C # D

FOUND
Waktu Pencarian : 0.132834ms
Banyak Kasus Yang Di Tinjau: 114

AABB
AABB

Load txt

Process

Optimized Process

Save Output

Output Interval in Cases (Higher is lower)

100000

Output Interval in Milliseconds

3000

N != M

3. ↵

AB
CC

Load txt

Process

Optimized Process

Save Output

Output Interval in Cases (Higher is lower)

100000

Output Interval in Milliseconds

3000.

Jumlah warna != N

AAAA
BBBB
CCCC
DDD

Load txt

Process

Optimized Process

Save Output

Output Interval in Cases (Higher is lower)

100000

Output Interval in Milliseconds

3000.

Row tidak lengkap

5. ↵

4.Pranala

https://github.com/Ahmad-FP/Tucil1_13524141

5. Pernyataan

Tugas ini disusun sepenuhnya tanpa bantuan kecerdasan buatan (Generative AI), melainkan hasil pemikiran dan analisis mandiri.



Ahmad Fauzan Putra