

# **Laporan Tugas Kecil 1**

## **IF2211/Strategi Algoritma**

**Penyelesaian Permainan Kartu 24 dengan Algoritma  
Brute Force**



**Disusun Oleh:**

**Muhammad Equilibrie Fajria - 13521047**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**BANDUNG**

**2023**

# 1. Algoritma Brute Force

Untuk menyelesaikan persoalan permainan kartu 24, digunakan algoritma brute force. Algoritma brute force merupakan algoritma yang *straightforward* dalam menyelesaikan persoalan. Secara singkat, cara kerja algoritma yang digunakan pada persoalan ini yaitu program akan mencoba semua kemungkinan yang ada berdasarkan masukan lalu apabila kemungkinan tersebut memenuhi persyaratan (hasilnya 24), maka kemungkinan tersebut akan disimpan sebagai solusi. Berikut adalah penjelasan langkah-langkah algoritma yang digunakan secara detail:

1. Program akan menanyakan pengguna, apakah ingin menginput manual atau dirandom. Jika pengguna memilih menginput manual, maka pengguna akan diminta memasukkan 4 angka/huruf sesuai dengan masukan yang valid (A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K). Pengguna akan diminta menginput kembali jika masukan salah.
2. Setelah mendapatkan input, program akan mengubah input dari string ke double.
3. Program lalu mencari solusi dengan menggunakan method `searchSolution`. Pada `searchSolution`, setiap urutan masukan yang mungkin, dicari kombinasinya dengan operator penjumlahan, pengurangan, perkalian, pembagian, dan tanda kurung dengan menggunakan method `singleSolution`. Jika hasil operasi dari keempat masukan sama dengan dua puluh empat, maka solusi tersebut akan disimpan pada `Set<String> result`.
4. Setelah semua kombinasi dicari, `searchSolution` mengembalikan `Set<String>` yang berisikan semua solusi yang didapat. Program lalu menampilkan semua solusi pada `Set<String>` tersebut dan menanyakan apakah pengguna ingin menyimpan solusi atau tidak. Jika pengguna memilih untuk menyimpan solusi, maka pengguna akan diminta nama file untuk menyimpan solusi tersebut.
5. Pada bagian akhir, program akan menanyakan pengguna apakah akan keluar program atau tidak. Jika pengguna memilih iya, maka program selesai. Jika pengguna memilih tidak, maka program akan mulai dari langkah pertama lagi.

## 2. Source Code

```
import java.io.FileWriter;
import java.io.IOException;
import java.util.*;

public class Main {
    public static void main(String[] args) throws IOException {
        Scanner input = new Scanner(System.in);
        String a, b, c, d;

        System.out.println("Selamat datang di game 24 solver\n");
        System.out.println("Silahkan pilih metode input:");
        System.out.println("1. Input manual");
        System.out.println("2. Random");
        String inputMethod = input.next();
        while (!(inputMethod.equals("1") || inputMethod.equals("2"))) {
            System.out.println("Pilihan tidak sesuai\n");
            System.out.println("Pilih metode input:");
            System.out.println("1. Input manual");
            System.out.println("2. Random");
            inputMethod = input.next();
        }

        // Menerima input manual
        if (inputMethod.equals("1")) {
            System.out.println("Silahkan masukkan input: ");
            a = input.next();
            b = input.next();
            c = input.next();
            d = input.next();
            while (invalidInput(a) || invalidInput(b) ||
invalidInput(c) || invalidInput(d)) {
                System.out.println("Masukan tidak sesuai");
                a = input.next();
                b = input.next();
                c = input.next();
                d = input.next();
            }
        }
        // Menerima input random
        else {
            String[] inputList = {"A", "2", "3", "4", "5", "6", "7",
"8", "9", "10", "J", "Q", "K"};
            Random rand = new Random();
            a = inputList[rand.nextInt(inputList.length)];
            b = inputList[rand.nextInt(inputList.length)];
            c = inputList[rand.nextInt(inputList.length)];
            d = inputList[rand.nextInt(inputList.length)];
        }
        System.out.printf("Input anda = %s %s %s %s\n\n", a, b, c, d);

        // Mengubah input dari string ke double
        double var1 = strToDouble(a);
        double var2 = strToDouble(b);
        double var3 = strToDouble(c);
        double var4 = strToDouble(d);

        // Mencari solusi
        long start = System.nanoTime();
        Set<String> solution = new HashSet<>(searchSolution(var1, var2,
```

```

var3, var4));
    long end = System.nanoTime();
    long executionTime = end - start;
    int nSolution = solution.size();

    if (nSolution == 0) {
        System.out.println("Tidak ada solusi");
    }
    else {
        System.out.printf("%d solusi ditemukan%n", nSolution);
        for (String s : solution) {
            System.out.println(s);
        }
    }
    System.out.printf("Waktu eksekusi: %f milliseconds%n%n",
executionTime/1000000.0);
    System.out.println("Apakah ingin menyimpan solusi?");
    System.out.println("1. Ya");
    System.out.println("2. Tidak");

    String save = input.next();
    while (!(save.equals("1") || save.equals("2"))) {
        System.out.println("Pilihan tidak sesuai\n");
        System.out.println("Apakah ingin menyimpan solusi?");
        System.out.println("1. Ya");
        System.out.println("2. Tidak");
        save = input.next();
    }

    // Menyimpan solusi
    if (save.equals("1")) {
        System.out.println("Ketik nama file untuk menyimpan solusi:
");
        String fileName = input.next();
        FileWriter writer = new FileWriter(String.format("%s.txt",
fileName));
        for (String str: solution) {
            writer.write(str + System.lineSeparator());
        }
        writer.close();
    }
    System.out.println("Keluar program?");
    System.out.println("1. Ya");
    System.out.println("2. Tidak");

    String exit = input.next();
    while (!(exit.equals("1") || exit.equals("2"))) {
        System.out.println("Pilihan tidak sesuai\n");
        System.out.println("Keluar program?");
        System.out.println("1. Ya");
        System.out.println("2. Tidak");
        exit = input.next();
    }
    if (exit.equals("2")) {
        main(args);
    }
}

public static boolean invalidInput (String var) {
    return !var.equals("A") && !var.equals("2") && !var.equals("3")
&& !var.equals("4") && !var.equals("5") && !var.equals("6")

```

```

    && !var.equals("7") && !var.equals("8") && !var.equals("9")
    && !var.equals("10") && !var.equals("J") && !var.equals("Q")
    && !var.equals("K");
    }

    public static double strToDouble (String var) {
        return switch (var) {
            case "A" -> 1;
            case "J" -> 11;
            case "Q" -> 12;
            case "K" -> 13;
            default -> Double.parseDouble(var);
        };
    }

    public static Set<String> searchSolution (double var1, double var2,
double var3, double var4) {
        Set<String> result = new HashSet<>();
        List<Double> input = new ArrayList<>();
        input.add(var1);
        input.add(var2);
        input.add(var3);
        input.add(var4);
        List<Double> input1;
        List<Double> input2;
        List<Double> input3;
        double a, b, c, d;

        for (int i = 0; i < input.size(); i++) {
            a = input.get(i);
            input1 = new ArrayList<>(input);
            input1.remove(i);
            for (int j = 0; j < input1.size(); j++) {
                b = input1.get(j);
                input2 = new ArrayList<>(input1);
                input2.remove(j);
                for (int k = 0; k < input2.size(); k++) {
                    c = input2.get(k);
                    input3 = new ArrayList<>(input2);
                    input3.remove(k);
                    d = input3.get(0);
                    result.addAll(singleSolution(a,b,c,d));
                }
            }
        }
        return result;
    }

    public static Set<String> singleSolution (double var1, double var2,
double var3, double var4) {
        Set<String> result = new HashSet<>();
        double firstOp1, secOp1, finalOp1;
        double firstOp2, secOp2, finalOp2;
        double firstOp3, secOp3, finalOp3;
        double firstOp4, secOp4, finalOp4;

        for (int i = 0; i < 4; i++) {
            firstOp1 = oneOperation(i,var1,var2);
            firstOp2 = oneOperation(i,var2,var3);
            firstOp3 = firstOp1;
            firstOp4 = firstOp2;

```

```

        for (int j = 0; j < 4; j++) {
            secOp1 = oneOperation(j, firstOp1, var3);
            secOp2 = oneOperation(j, var1, firstOp2);
            secOp3 = oneOperation(j, var3, var4);
            secOp4 = oneOperation(j, firstOp4, var4);
            for (int k = 0; k < 4; k++) {
                finalOp1 = oneOperation(k, secOp1, var4);
                finalOp2 = oneOperation(k, secOp2, var4);
                finalOp3 = oneOperation(k, firstOp3, secOp3);
                finalOp4 = oneOperation(k, var1, secOp4);
                // ((var1 Op var2) Op var3) Op var4
                if (Math.abs(finalOp1 - 24) < 0.0001) {
result.add(solutionToString(i, j, k, 1, var1, var2, var3, var4));
                }
                // (var1 Op (var2 Op var3)) Op var4
                if (Math.abs(finalOp2 - 24) < 0.0001) {
result.add(solutionToString(j, i, k, 2, var1, var2, var3, var4));
                }
                // (var1 Op var2) Op (var3 Op var4)
                if (Math.abs(finalOp3 - 24) < 0.0001) {
result.add(solutionToString(i, k, j, 3, var1, var2, var3, var4));
                }
                // var1 Op ((var2 Op var3) Op var4)
                if (Math.abs(finalOp4 - 24) < 0.0001) {
result.add(solutionToString(k, i, j, 4, var1, var2, var3, var4));
                }
            }
        }
    }
    return result;
}

public static double oneOperation (int i, double var1, double var2)
{
    if (i == 0) {
        return (var1 + var2);
    }
    else if (i == 1) {
        return (var1 - var2);
    }
    else if (i == 2) {
        return (var1 * var2);
    }
    else {
        if (var2 == 0) {
            return (-999);
        }
        else {
            return (var1 / var2);
        }
    }
}

public static String solutionToString (int i, int j, int k, int l,
double a, double b, double c, double d) {
    String result;
    long var1 = Math.round(a);

```

```

        long var2 = Math.round(b);
        long var3 = Math.round(c);
        long var4 = Math.round(d);

        if (l == 1) {
            result = String.format("(%d %s %d) %s %d) %s %d", var1,
operatorToString(i), var2, operatorToString(j), var3,
operatorToString(k), var4);
        }
        else if (l == 2) {
            result = String.format("(%d %s (%d %s %d)) %s %d", var1,
operatorToString(i), var2, operatorToString(j), var3,
operatorToString(k), var4);
        }
        else if (l == 3) {
            result = String.format("(%d %s %d) %s (%d %s %d)", var1,
operatorToString(i), var2, operatorToString(j), var3,
operatorToString(k), var4);
        }
        else {
            result = String.format("%d %s ((%d %s %d) %s %d)", var1,
operatorToString(i), var2, operatorToString(j), var3,
operatorToString(k), var4);
        }
        return result;
    }

    public static String operatorToString (int i) {
        return switch (i) {
            case (0) -> "+";
            case (1) -> "-";
            case (2) -> "*";
            default -> "/";
        };
    }
}

```

### 3. Test Case

#### a. Input tidak valid

```
Silahkan pilih metode input:
1. Input manual
2. Random
4
Pilihan tidak sesuai
```

```
Pilih metode input:
1. Input manual
2. Random
a
Pilihan tidak sesuai
```

```
Silahkan pilih metode input:
1. Input manual
2. Random
1
Silahkan masukkan input:
C F @ P
Masukan tidak sesuai
```

```
Keluar program?
1. Ya
2. Tidak
3
Pilihan tidak sesuai

Keluar program?
1. Ya
2. Tidak
t
Pilihan tidak sesuai
```

```
Silahkan pilih metode input:
1. Input manual
2. Random
1
Silahkan masukkan input:
1 1 1 1
Masukan tidak sesuai
```

```
Silahkan pilih metode input:
1. Input manual
2. Random
1
Silahkan masukkan input:
13 12 Q K
Masukan tidak sesuai
```

```
Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
3
Pilihan tidak sesuai

Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
k
Pilihan tidak sesuai
```



## b. Input Valid

### 1. Tes 1

Selamat datang di game 24 solver

Silahkan pilih metode input:

1. Input manual
2. Random

1

Silahkan masukkan input:

7 4 3 7

Input anda = 7 4 3 7

40 solusi ditemukan

$(7 - (1 - 7)) + 11$

$((7 + 11) + 7) - 1$

$(7 + 7) + (11 - 1)$

$11 - ((1 - 7) - 7)$

$(7 + 11) + (7 - 1)$

$(7 + (11 + 7)) - 1$

$((11 + 7) - 1) + 7$

$(7 - 1) * (11 - 7)$

$(7 - 11) * (1 - 7)$

$11 + ((7 - 1) + 7)$

$(11 - 7) * (7 - 1)$

$((7 - 1) + 7) + 11$

$((11 - 1) + 7) + 7$

$11 + ((7 + 7) - 1)$

$(11 + 7) + (7 - 1)$

$(7 + 11) - (1 - 7)$

$(1 - 7) * (7 - 11)$

$7 - ((1 - 11) - 7)$

$(11 + 7) - (1 - 7)$

$((7 + 7) + 11) - 1$

$((7 + 7) - 1) + 11$

$(7 - 1) + (7 + 11)$

$7 + ((7 - 1) + 11)$

$7 + ((11 + 7) - 1)$

$(7 - 1) + (11 + 7)$

$((7 - 1) + 11) + 7$

$(7 - (1 - 11)) + 7$

$7 + ((7 + 11) - 1)$

$(11 + (7 + 7)) - 1$

$(11 - 1) + (7 + 7)$

$(7 + (11 - 1)) + 7$

$7 - ((1 - 7) - 11)$

$(7 + (7 - 1)) + 11$

$(7 + (7 + 11)) - 1$

$((11 + 7) + 7) - 1$

$7 + ((11 - 1) + 7)$

$((7 + 11) - 1) + 7$

$(11 - (1 - 7)) + 7$

$(11 + (7 - 1)) + 7$

$(7 + 7) - (1 - 11)$

Waktu eksekusi: 6.166300 milliseconds

Apakah ingin menyimpan solusi?

1. Ya
2. Tidak

1

Ketik nama file untuk menyimpan solusi:

tes1

Keluar program?

1. Ya
2. Tidak

1

Process finished with exit code 0

tes1.txt - Notepad

File Edit View

```
| (7 - (1 - 7)) + 11
| ((7 + 11) + 7) - 1
| (7 + 7) + (11 - 1)
| 11 - ((1 - 7) - 7)
| (7 + 11) + (7 - 1)
| (7 + (11 + 7)) - 1
| ((11 + 7) - 1) + 7
| (7 - 1) * (11 - 7)
| (7 - 11) * (1 - 7)
| 11 + ((7 - 1) + 7)
| (11 - 7) * (7 - 1)
| ((7 - 1) + 7) + 11
| ((11 - 1) + 7) + 7
| 11 + ((7 + 7) - 1)
| (11 + 7) + (7 - 1)
| (7 + 11) - (1 - 7)
| (1 - 7) * (7 - 11)
| 7 - ((1 - 11) - 7)
| (11 + 7) - (1 - 7)
| ((7 + 7) + 11) - 1
| ((7 + 7) - 1) + 11
| (7 - 1) + (7 + 11)
| 7 + ((7 - 1) + 11)
| 7 + ((11 + 7) - 1)
| (7 - 1) + (11 + 7)
| ((7 - 1) + 11) + 7
| (7 - (1 - 11)) + 7
| 7 + ((7 + 11) - 1)
| (11 + (7 + 7)) - 1
| (11 - 1) + (7 + 7)
| (7 + (11 - 1)) + 7
| 7 - ((1 - 7) - 11)
| (7 + (7 - 1)) + 11
| (7 + (7 + 11)) - 1
| ((11 + 7) + 7) - 1
| 7 + ((11 - 1) + 7)
| ((7 + 11) - 1) + 7
| (11 - (1 - 7)) + 7
| (11 + (7 - 1)) + 7
| (7 + 7) - (1 - 11)
```

## 2. Tes 2

Selamat datang di game 24 solver

Silahkan pilih metode input:

1. Input manual
2. Random

}

Silahkan masukkan input:

A J Q K

Input anda = A J Q K

26 solusi ditemukan

12 \* ((13 \* 1) - 11)  
(13 - 11) \* (12 \* 1)  
(13 - (11 \* 1)) \* 12  
((13 / 1) - 11) \* 12  
((13 - 11) / 1) \* 12  
(13 - 11) / (1 / 12)  
(12 \* 1) \* (13 - 11)  
12 \* ((13 / 1) - 11)  
(13 - (1 \* 11)) \* 12  
((13 \* 1) - 11) \* 12  
(13 - 11) \* (1 \* 12)  
1 \* ((13 - 11) \* 12)  
12 \* ((1 \* 13) - 11)  
((13 - 11) \* 12) / 1  
12 \* ((13 - 11) \* 1)  
(1 \* (13 - 11)) \* 12  
(12 \* (13 - 11)) \* 1

((1 \* 13) - 11) \* 12  
(13 - 11) \* (12 / 1)  
(12 \* (13 - 11)) / 1  
((13 - 11) \* 12) \* 1  
(12 / 1) \* (13 - 11)  
((13 - 11) \* 1) \* 12  
(1 \* 12) \* (13 - 11)  
12 \* ((13 - 11) / 1)  
(13 - (11 / 1)) \* 12

Waktu eksekusi: 3.416700 milliseconds

Apakah ingin menyimpan solusi?

1. Ya
2. Tidak

}

Ketik nama file untuk menyimpan solusi:

tes2

Keluar program?

1. Ya
2. Tidak

}

Process finished with exit code 0



tes2.txt - Notepad

File Edit View

12 \* ((13 \* 1) - 11)  
(13 - 11) \* (12 \* 1)  
(13 - (11 \* 1)) \* 12  
((13 / 1) - 11) \* 12  
((13 - 11) / 1) \* 12  
(13 - 11) / (1 / 12)  
(12 \* 1) \* (13 - 11)  
12 \* ((13 / 1) - 11)  
(13 - (1 \* 11)) \* 12  
((13 \* 1) - 11) \* 12  
(13 - 11) \* (1 \* 12)  
1 \* ((13 - 11) \* 12)  
12 \* ((1 \* 13) - 11)  
((13 - 11) \* 12) / 1  
12 \* ((13 - 11) \* 1)  
(1 \* (13 - 11)) \* 12  
(12 \* (13 - 11)) \* 1  
(12 \* (13 - 11)) \* 1  
(1 \* 12) \* (13 - 11)  
12 \* ((13 - 11) / 1)  
(13 - (11 / 1)) \* 12

### 3. Tes 3

Selamat datang di game 24 solver

Silahkan pilih metode input:

1. Input manual
2. Random

1

Silahkan masukkan input:

2 2 2 2

Input anda = 2 2 2 2

Tidak ada solusi

Waktu eksekusi: 1.215300 milliseconds

Apakah ingin menyimpan solusi?

1. Ya
2. Tidak

1

Ketik nama file untuk menyimpan solusi:

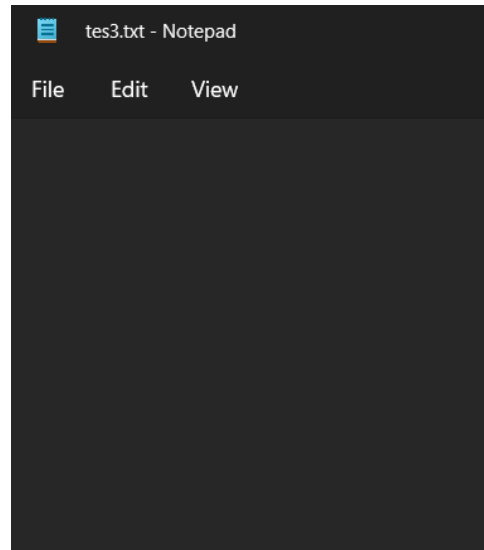
tes3

Keluar program?

1. Ya
2. Tidak

1

Process finished with exit code 0



## 4. Tes 4

```
Selamat datang di game 24 solver

Silahkan pilih metode input:
1. Input manual
2. Random
>
Input anda = 2 5 8 4

47 solusi ditemukan
(5 * (8 / 2)) + 4
((8 - 5) * 4) * 2
(2 + (5 - 4)) * 8
((5 + 2) - 4) * 8
8 * ((5 - 4) + 2)
4 + ((5 * 8) / 2)
(8 / 2) + (4 * 5)
(4 * (8 - 5)) * 2
(2 - (4 - 5)) * 8
(5 / (2 / 8)) + 4
2 * ((8 - 5) * 4)
4 + ((8 * 5) / 2)
8 * ((4 * 2) - 5)
(8 * (5 / 2)) + 4
8 * ((2 * 4) - 5)
(5 - (4 / 2)) * 8
(5 + (2 - 4)) * 8
(4 * 5) + (8 / 2)
8 * ((2 - 4) + 5)
```

```
8 * ((5 + 2) - 4)
2 * ((5 * 4) - 8)
((5 * 8) / 2) + 4
2 * ((4 * 5) - 8)
(8 - 5) * (4 * 2)
((4 * 2) - 5) * 8
8 * ((2 + 5) - 4)
((8 / 2) * 5) + 4
((2 + 5) - 4) * 8
((5 / 2) * 8) + 4
((5 - 4) + 2) * 8
4 + ((8 / 2) * 5)
(5 - (4 - 2)) * 8
((8 - 5) * 2) * 4
((2 * 4) - 5) * 8
((4 * 5) - 8) * 2
((2 - 4) + 5) * 8
(8 / (2 / 5)) + 4
(2 * (8 - 5)) * 4
4 * ((8 - 5) * 2)
(8 - 5) * (2 * 4)
((8 * 5) / 2) + 4
(5 * 4) + (8 / 2)
((5 * 4) - 8) * 2
(4 * 2) * (8 - 5)
(8 / 2) + (5 * 4)
4 + ((5 / 2) * 8)
(2 * 4) * (8 - 5)

Waktu eksekusi: 5.232700 milliseconds
```

```
Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
1
Ketik nama file untuk menyimpan solusi:
tes4
Keluar program?
1. Ya
2. Tidak
1
Process finished with exit code 0
```

```
tes4.txt - Notepad

File Edit View

(5 * (8 / 2)) + 4
((8 - 5) * 4) * 2
(2 + (5 - 4)) * 8
((5 + 2) - 4) * 8
8 * ((5 - 4) + 2)
4 + ((5 * 8) / 2)
(8 / 2) + (4 * 5)
(4 * (8 - 5)) * 2
(2 - (4 - 5)) * 8
(5 / (2 / 8)) + 4
2 * ((8 - 5) * 4)
4 + ((8 * 5) / 2)
8 * ((4 * 2) - 5)
(8 * (5 / 2)) + 4
8 * ((2 * 4) - 5)
(5 - (4 / 2)) * 8
(5 + (2 - 4)) * 8
(4 * 5) + (8 / 2)
8 * ((2 - 4) + 5)
8 * ((5 + 2) - 4)
2 * ((5 * 4) - 8)
((5 * 8) / 2) + 4
2 * ((4 * 5) - 8)
(8 - 5) * (4 * 2)
((4 * 2) - 5) * 8
8 * ((2 + 5) - 4)
((8 / 2) * 5) + 4
((2 + 5) - 4) * 8
((5 / 2) * 8) + 4
((5 - 4) + 2) * 8
4 + ((8 / 2) * 5)
(5 - (4 - 2)) * 8
((8 - 5) * 2) * 4
((2 * 4) - 5) * 8
((4 * 5) - 8) * 2
((2 - 4) + 5) * 8
(8 / (2 / 5)) + 4
(2 * (8 - 5)) * 4
4 * ((8 - 5) * 2)
(8 - 5) * (2 * 4)
((8 * 5) / 2) + 4
(5 * 4) + (8 / 2)
((5 * 4) - 8) * 2
(4 * 2) * (8 - 5)
(8 / 2) + (5 * 4)
4 + ((5 / 2) * 8)
(2 * 4) * (8 - 5)
```

## 5. Tes 5

Selamat datang di game 24 solver

Silahkan pilih metode input:

1. Input manual
2. Random

2

Input anda = K J K 6

Tidak ada solusi

Waktu eksekusi: 1.589100 milliseconds

Apakah ingin menyimpan solusi?

1. Ya
2. Tidak

1

Ketik nama file untuk menyimpan solusi:

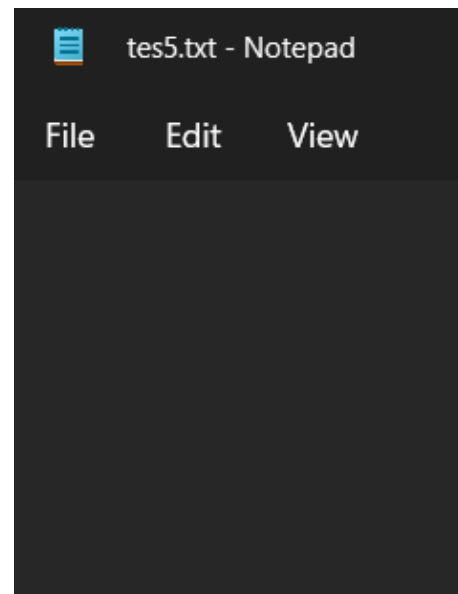
tes5

Keluar program?

1. Ya
2. Tidak

1

Process finished with exit code 0



## 6. Tes 6

Selamat datang di game 24 solver

Silahkan pilih metode input:

1. Input manual
2. Random

2

Input anda = J A 8 K

12 solusi ditemukan

$(13 - (11 - 1)) * 8$   
 $(1 + (13 - 11)) * 8$   
 $(1 - (11 - 13)) * 8$   
 $8 * ((13 + 1) - 11)$   
 $((1 + 13) - 11) * 8$   
 $8 * ((1 - 11) + 13)$   
 $(13 + (1 - 11)) * 8$   
 $((13 + 1) - 11) * 8$   
 $((1 - 11) + 13) * 8$   
 $8 * ((1 + 13) - 11)$   
 $8 * ((13 - 11) + 1)$   
 $((13 - 11) + 1) * 8$

Waktu eksekusi: 3.692400 milliseconds

tes6.txt - Notepad

File Edit View

$(13 - (11 - 1)) * 8$   
 $(1 + (13 - 11)) * 8$   
 $(1 - (11 - 13)) * 8$   
 $8 * ((13 + 1) - 11)$   
 $((1 + 13) - 11) * 8$   
 $8 * ((1 - 11) + 13)$   
 $(13 + (1 - 11)) * 8$   
 $((13 + 1) - 11) * 8$   
 $((1 - 11) + 13) * 8$   
 $8 * ((1 + 13) - 11)$   
 $8 * ((13 - 11) + 1)$   
 $((13 - 11) + 1) * 8$

Apakah ingin menyimpan solusi?

1. Ya
2. Tidak

1

Ketik nama file untuk menyimpan solusi:

tes6

Keluar program?

1. Ya
2. Tidak

1

Process finished with exit code 0

7. Tes 7

```
Selamat datang di game 24 solver

Silahkan pilih metode input:
1. Input manual
2. Random
3

Input anda = Q 8 6 2

96 solusi ditemukan
(12 + (8 - 2)) + 6
(6 * (8 - 2)) - 12
(8 - (2 - 12)) + 6
6 * ((8 * 2) - 12)
(6 - 2) + (12 + 8)
(8 + 12) - (2 - 6)
((8 - 2) + 12) + 6
((8 + 12) + 6) - 2
6 + ((12 - 2) + 8)
8 * ((12 - 6) / 2)
(8 * 6) - (12 * 2)
8 - ((2 - 12) - 6)
(8 + (6 - 2)) + 12
((6 + 8) + 12) - 2
((6 - 2) + 8) + 12
((6 - 2) + 12) + 8
((8 * 2) - 12) * 6
12 + ((6 + 8) - 2)
((12 - 6) * 8) / 2
12 + ((8 + 6) - 2)
```

$$\begin{array}{l}
(6 - (8 / 2)) * 12 \\
(8 * 12) / (6 - 2) \\
(8 + 6) - (2 - 12) \\
12 + ((8 - 2) + 6) \\
(12 + (6 - 2)) + 8 \\
(8 + (6 + 12)) - 2 \\
(6 + (12 - 2)) + 8 \\
(12 - (2 - 6)) + 8 \\
((8 + 6) + 12) - 2 \\
12 - ((2 - 8) - 6) \\
12 / ((6 - 2) / 8) \\
(12 + 8) - (2 - 6) \\
((8 - 2) * 6) - 12 \\
(6 * 8) - (12 * 2) \\
(6 + (8 - 2)) + 12 \\
((6 + 8) - 2) + 12 \\
((2 * 8) - 12) * 6 \\
((12 - 2) + 6) + 8 \\
6 + ((12 + 8) - 2) \\
(8 + 12) + (6 - 2) \\
(8 / (6 - 2)) * 12 \\
8 + ((6 - 2) + 12) \\
8 / ((6 - 2) / 12) \\
(12 - 6) * (8 / 2) \\
((12 + 8) - 2) + 6 \\
(8 + (12 - 2)) + 6 \\
(6 + (8 + 12)) - 2 \\
((12 - 6) / 2) * 8 \\
(8 * (12 - 6)) / 2
\end{array}$$
$$\begin{aligned} & (12 \div (6 - 2)) * 8 \\ & ((8 + 6) - 2) + 12 \\ & (8 * 6) - (2 * 12) \\ & ((12 + 6) - 2) + 8 \\ & 8 - ((2 - 6) - 12) \\ & (12 - 2) + (8 + 6) \\ & (6 + 8) - (2 - 12) \\ & 6 + ((8 - 2) + 12) \\ & (8 + 6) + (12 - 2) \\ & (12 + 6) + (8 - 2) \\ & 8 + ((12 - 2) + 6) \\ & (12 + 8) + (6 - 2) \\ & 6 - ((2 - 12) - 8) \\ & (8 \div 2) * (12 - 6) \\ & 6 * ((2 * 8) - 12) \\ & 8 + ((12 + 6) - 2) \\ & (6 - (2 - 12)) + 8 \\ & (8 - 2) + (12 + 6) \\ & (12 - 2) + (6 + 8) \\ & ((12 + 8) + 6) - 2 \\ & (8 - (2 - 6)) + 12 \\ & 6 + ((8 + 12) - 2) \\ & (12 * 8) \div (6 - 2) \\ & 12 + ((6 - 2) + 8) \\ & (12 - 6) \div (2 \div 8) \\ & 6 - ((2 - 8) - 12) \\ & (6 + 12) - (2 - 8) \\ & 12 - ((2 - 6) - 8) \\ & ((6 + 12) - 2) + 8 \end{aligned}$$

```
( (8 + 12) - 2) + 6
(6 * 8) - (2 * 12)
((12 + 6) + 8) - 2
((8 - 2) + 6) + 12
(6 + 12) + (8 - 2)
(6 - (2 - 8)) + 12
(12 + (6 + 8)) - 2
(6 + (12 + 8)) - 2
(6 + 8) + (12 - 2)
(12 + (8 + 6)) - 2
(12 + 6) - (2 - 8)
((12 - 2) + 8) + 6
(8 + (12 + 6)) - 2
8 + ((6 + 12) - 2)
(12 - (2 - 8)) + 6
(8 - 2) + (6 + 12)
((6 + 12) + 8) - 2
(6 - 2) + (8 + 12)
```

Waktu eksekusi: 7.029500 milliseconds

```
Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
1
Ketik nama file untuk menyimpan solusi:
tes2
Keluar program?
1. Ya
2. Tidak
1
Process finished with exit code 0
```

```

tes7.txt - Notepad
File Edit View

(12 + (8 - 2)) + 6
(6 * (8 - 2)) - 12
(8 - (2 - 12)) + 6
6 * ((8 * 2) - 12)
(6 - 2) + (12 + 8)
(8 + 12) - (2 - 6)
((8 - 2) + 12) + 6
((8 + 12) + 6) - 2
6 + ((12 - 2) + 8)
8 * ((12 - 6) / 2)
(8 * 6) - (12 * 2)
8 - ((2 - 12) - 6)
(8 + (6 - 2)) + 12
((6 + 8) + 12) - 2
((6 - 2) + 8) + 12
((6 - 2) + 12) + 8
((8 * 2) - 12) * 6
12 + ((6 + 8) - 2)
((12 - 6) * 8) / 2
12 + ((8 + 6) - 2)
(12 / (6 - 2)) * 8
((8 + 6) - 2) + 12
(8 * 6) - (2 * 12)
((12 + 6) - 2) + 8
8 - ((2 - 6) - 12)
((12 - 2) + (8 + 6))
(6 + 8) - (2 - 12)
6 + ((8 - 2) + 12)
(8 + 6) + (12 - 2)
((12 + 6) + (8 - 2))
8 + ((12 - 2) + 6)
((12 + 8) + (6 - 2))
6 - ((2 * 12) - 8)
(8 / 2) * (12 - 6)
8 * ((2 * 8) - 12)
8 + ((12 + 6) - 2)
(6 - (2 - 12)) + 8
(8 - 2) + (12 + 6)
((12 - 2) + (6 + 8))
((12 + 8) + 6) - 2
8 - (2 - 6)
((8 + 12) - 2) + 12
(12 * 8) / (6 - 2)
12 + ((6 - 2) + 8)
((12 - 6) / (2 - 1))
6 - ((2 - 8) - 12)
(6 + 12) - (2 - 8)
12 - ((2 - 6) - 8)
((6 + 12) - 2) + 8
(6 - (8 / 2)) + 12
(8 * 12) / (6 - 2)
(8 + 6) - (2 - 12)
12 + ((8 - 2) + 6)
((12 + (6 - 2)) + 8)
(8 + (6 + 12)) + 1
(6 + (12 - 2)) + 1
(12 - (2 - 6)) + 1
(8 + 6) + 12
12 - ((2 - 8) * 6)
12 / ((6 - 2) - 1)
(6 * 8) - (12 * 2)
(6 + 8) - (2 * 12)
(3 * 8) - 12
((12 - 2) + 6) + 8
8 + ((12 + 8) + 6)
(8 + 12) + (6 - 2)
8 / ((6 - 2) + 1)
((6 + 8) * 6) / (2 - 1)
((6 + 8) - 2) + 12
((6 - 2) + 12) + 1
((6 - 2) + 12) / 1
(12 - 6) * (8 / 2)
((12 + 8) - 2) + 1
(6 + (12 - 2)) + 12
(8 + (8 + 12)) + 1
(12 * 6) / 2
(8 * (12 - 6)) - 2
(8 + 12) - (2 - 2)
(6 * 8) - (2 * 12)
((12 - 6) + 8) + 12
(6 - 2) + (6 + 8)
(6 + 12) + (8 + 1)
(6 - (2 - 8)) + 1
(12 + (6 + 8)) + 1
((12 + 8) + 12) + 1
((6 + 8) + (12 - 2)) - 2
(12 + (8 + 6)) - 2
((12 - 2) + (6 - 8)) - 1
8 - ((12 + 6) * 2)
(1 * - (6 - 12)) - 6
(8 - 2) * (8 - 1)
((6 + 12) + 8) + 1
(6 - 2) + (8 + 12)

```

## 8. Tes 8

```
Selamat datang di game 24 solver

Silahkan pilih metode input:
1. Input manual
2. Random
2
Input anda = Q 3 Q Q

1 solusi ditemukan
(3 - (12 / 12)) * 12
Waktu eksekusi: 1.211800 milliseconds

Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
1
Ketik nama file untuk menyimpan solusi:
tes8
Keluar program?
1. Ya
2. Tidak
1

Process finished with exit code 0
```

 tes8.txt - Notepad

File Edit View

(3 - (12 / 12)) \* 12



## 9. Tes 9

```
Selamat datang di game 24 solver

Silahkan pilih metode input:
1. Input manual
2. Random
2
Input anda = 3 J 5 9

Tidak ada solusi
Waktu eksekusi: 0.826800 miliseconds

Apakah ingin menyimpan solusi?
1. Ya
2. Tidak
2

Keluar program?
1. Ya
2. Tidak
1

Process finished with exit code 0
```

#### 4. Referensi

1. [https://www.youtube.com/watch?v=ET\\_HIyJTl1E](https://www.youtube.com/watch?v=ET_HIyJTl1E)
2. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-\(2022\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-(2022)-Bag1.pdf)
3. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-\(2022\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-(2022)-Bag2.pdf)

#### 5. Lampiran

Link repository: [https://github.com/MuhLibri/Tucil1\\_13521047.git](https://github.com/MuhLibri/Tucil1_13521047.git)

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan	✓	
2. Program berhasil <i>running</i>	✓	
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	✓	