

LAPORAN TUGAS KECIL 1
IF2211 STRATEGI ALGORITMA

Penyelesaian Permainan Kartu 24 dengan *Algoritma Brute Force*

Dosen Pengampu : Dr.Ir.Rinaldi, M.T.



Oleh :

Irgiansyah Mondo / 13521167

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG
2022/2023

1. Deskripsi Program

a. Deskripsi Permainan 24

Permainan 24 adalah permainan di mana pemain menggunakan 4 buah angka dan mencoba mencari operasi aritmatika yang menghubungkan keempat angka tersebut sehingga terbentuk sebuah angka 24.

Operasi aritmatika yang dapat digunakan dalam permainan tersebut adalah tambah (+), kurang (-), kali (x), dan bagi (/). Setelah pemain mendapatkan keempat angka, maka pemain harus mencari operasi yang mungkin menggunakan keempat operator tersebut. Tentu saja operasi dari keempat angka tersebut haruslah kurang dari

b. Deskripsi Algoritma Brute Force

Dalam kesempatan ini, mata kuliah Strategi Algoritma menugaskan untuk mengaplikasikan strategi brute force untuk mencari setiap solusi 24 yang mungkin jika diberikan 4 bilangan bulat nonnegatif. Strategi brute force adalah strategi di mana algoritma mengecek semua kemungkinan yang ada untuk memberikan solusi dari permasalahan.

Strategi brute force adalah strategi Strategi ini dinilai naif karena algoritma ini menyelesaikan persoalan langsung sesuai dengan permintaan serta langsung mengecek semua kemungkinan.

Namun, ada beberapa persoalan yang hanya dapat diselesaikan menggunakan brute force dan penggunaan strategi ini dinilai lebih mudah dari strategi lain.

3. Kemungkinan penempatan tanda kurung dalam operasi

Penempatan tanda kurung dalam operasi aritmatika bekerja dalam menentukan prioritas operasi yang dilakukan. Dari penempatan tanda kurung tersebut, terdapat 5 buah jenis solusi yang ada. Jika dimisalkan salah satu susunan bilangan tersebut adalah a, b, c, dan d serta operasi yang digunakan adalah 3 operasi penjumlahan, maka susunan tanda kurung tersebut adalah sebagai berikut.

a. $((a + b) + c) + d$

b. $a + (b + (c + d))$

c. $(a + b) + (c + d)$

d. $(a + (b + c)) + d$

e. $a + ((b + c) + d)$

d. Deskripsi Langkah Kerja Program

1. Pertama masukan dibaca dengan fungsi `baca()`.
2. Lalu 4 angka tersebut dimasukan ke array untuk diproses oleh fungsi `tambaharr()`.
4. Setiap kemungkinan akan diproses oleh fungsi `testangka()`.
5. Lalu dihitung oleh fungsi `hitung()`.
6. Jika berhasil mendapatkan angka 24 maka akan diproses oleh fungsi `output()`.

2. Source Code

```

1  #include <iostream>
2
3
4  using namespace std;
5
6  // GLOBAL VARIABLES
7  float a, b, c, d;
8  int temp = 0;
9
10 int e[25], f[25], g[25], h[25];
11 int n = 0;
12
13 void tambaharr(int a, int b, int c, int d);
14
15
16 void baca();
17
18
19 void testangka (int w, int x, int y, int z);
20
21
22 void hasil (int w, int x, int y, int z, int i, int j, int k);
23
24
25 float hitung (float a, float b, int masukan);
26
27
28 char output (int masukan);
29
30
31 // PROGRAM UTAMA
32 int main()
33 {
34     // KAMUS
35     int i;
36
37     // ALGORITMA
38
39
40     cout << "24 GAME" << endl << endl;
41     baca
42 ();
43
44
45
46

```

```

47
48     tambaharr(c, b, a, d);
49     tambaharr(c, b, d, a);
50     tambaharr(c, d, a, b);
51     tambaharr(c, d, b, a);
52     tambaharr(d, a, b, c);
53     tambaharr(d, a, c, b);
54     tambaharr(d, b, a, c);
55     tambaharr(d, b, c, a);
56     tambaharr(d, c, a, b);
57     tambaharr(d, c, b, a);
58     tambaharr(a, b, c, d);
59     tambaharr(a, b, d, c);
60     tambaharr(a, c, b, d);
61     tambaharr(a, c, d, b);
62     tambaharr(a, d, b, c);
63     tambaharr(a, d, c, b);
64     tambaharr(b, a, c, d);
65     tambaharr(b, a, d, c);
66     tambaharr(b, c, a, d);
67     tambaharr(b, c, d, a);
68     tambaharr(b, d, a, c);
69     tambaharr(b, d, c, a);
70     tambaharr(c, a, b, d);
71     tambaharr(c, a, d, b);
72
73     for (i = 1; i <= n; i++)
74     {
75         testangka
76         (e[i], f[i], g[i], h[i]);
77     }
78
79
80
81     cout << endl << "Hasil = " << temp << " buah solusi" << endl;
82
83
84     return 0;
85 }
86
87

```

```

void baca()
{
    cout << "Masukkan Angka :" << endl;
    cin >> a >> b >> c >> d;
    cout << "diproses..." << endl;
    cout << a << ", " << b << ", " << c << ", dan " << d << endl << endl;
}

void tambaharr(int a, int b, int c, int d)
{
    // Kamus
    int i;
    bool tepat;

    // ALGORITMA
    tepat = false;
    i = 1;

    while (i <= n && not tepat)
    {
        if (a == e[i] && b == f[i] && c == g[i] && d == h[i])
        {
            tepat = true;
        }
        else
        {
            i++;
        }
    }

    if (not tepat)
    {
        n++;
        e[n] = a;
        f[n] = b;
        g[n] = c;
        h[n] = d;
    }
}

```

```

131
132 void testangka (int w, int x, int y, int z)
133 {
134     int o, p, q;
135
136     for (o = 1; o <= 4; o++)
137     {
138         for (p = 1; p <= 4; p++)
139         {
140             for (q = 1; q <= 4; q++)
141             {
142                 hasil (w, x, y, z, o, p, q);
143             }
144         }
145     }
146 }
147
148
149
150 void hasil (int w, int x, int y, int z, int i, int j, int k)
151 {
152     // Float ini berfungsi menyimpan hasil perhitungan
153     float a;
154
155
156     a = hitung(hitung(hitung((float)w, (float)x, i), (float) y, j), (float)z, k);
157     if (a > 23.99999 && a < 24.00001)
158     {
159         temp++;
160         cout << "(" << w << " " << output(i) << " " << x << ")" <<
161         | output(j) << " " << y << ")" << output(k) << " " << z << endl;
162     }
163
164     a = hitung(hitung((float)w, (float)x, i), hitung((float) y, (float) z, k), j);
165     if (a > 23.99999 && a < 24.00001)
166     {
167         temp++;
168         cout << "(" << w << " " << output(i) << " " << x << ")" <<
169         | output(j) << " (" << y << " " << output(k) << " " << z << ")" << endl;
170     }
171

```

```

171
172     a = hitung(hitung((float)w, hitung((float)x, (float)y, j), i), (float) z, k);
173     if (a > 23.99999 && a < 24.00001)
174     {
175         temp++;
176         cout << "(" << w << " " << output(i) << " (" << x << " " << output(j) <<
177         | " " << y << ") " << output(k) << " " << z << endl;
178     }
179
180     a = hitung((float)w, hitung(hitung((float)x, (float)y, j), (float)z, k), i);
181     if (a > 23.99999 && a < 24.00001)
182     {
183         temp++;
184         cout << w << " " << output(i) << " (" << x << " " << output(j) << " " <<
185         | y << ") " << output(k) << " " << z << ")" << endl;
186     }
187
188     a = hitung((float)w, hitung((float)x, hitung((float)y, (float)z, k), j), i);
189     if (a > 23.99999 && a < 24.00001)
190     {
191         temp++;
192         cout << w << " " << output(i) << " (" << x << " " << output(j) << " (" <<
193         | y << " " << output(k) << " " << z << ")") << endl;
194     }
195 }
196
197
198 float hitung (float a, float b, int masukan)
199 {
200     if (masukan == 1)
201     {
202         return (a + b);
203     }
204     else if (masukan == 2)
205     {
206         return (a - b);
207     }
208     else if (masukan == 3)
209     {
210         return (a * b);
211     }
212     else if (masukan == 4)
213     {
214         return ((float)a / (float)b);
215     }
216 }
217

```



```

218
219 char output (int masukan)
220 {
221     if (masukan == 1)
222     {
223         return ('+');
224     }
225     if (masukan == 2)
226     {
227         return ('-');
228     }
229     if (masukan == 3)
230     {
231         return ('*');
232     }
233     if (masukan == 4)
234     {
235         return ('/');
236     }
237 }

```

3. Tes Kasus

a.

```

PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucil1_13521167\src> cd "c:\Users\IRGIANSYAH\OneDrive\Documents\Tucil1_13521167\src\" ; if ($?) { g++ src.cpp -o src } ; if ($?) { .\src }
24 GAME

Masukkan Angka :
2 2 2 2
diproses...
2, 2, 2, dan 2

Hasil = 0 buah solusi
PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucil1_13521167\src>

```

Kasus : 2 2 2 2

Hasil : 0 solusi

b.

```

PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucil1_13521167\src> cd "c:\Users\IRGIANSYAH\OneDrive\Documents\Tucil1_13521167\src\" ; if ($?) { g++ src.cpp -o src } ; if ($?) { .\src }
24 GAME

Masukkan Angka :
3 4 7 8
diproses...
3, 4, 7, dan 8

8 - ((3 - 7) * 4)
8 - (4 * (3 - 7))
8 + (4 * (7 - 3))
8 + ((7 - 3) * 4)
(4 * (7 - 3)) + 8
((7 - 3) * 4) + 8

Hasil = 6 buah solusi

```

Kasus : 3 4 7 8

Hasil : 6 solusi

c.

```
Hasil = 0 buah solusi
PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src> cd "c:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src\" ; if ($?) { g++ src.cpp -o src } ; if ($?) { .\src }
24 GAME

Masukkan Angka :
9 9 9 9
diproses...
9, 9, 9, dan 9

Hasil = 0 buah solusi
```

Kasus : 9 9 9 9

Hasil : 0 solusi

d.

```
PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src> cd "c:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src\" ; if ($?) { g++ src.cpp -o src } ; if ($?) { .\src }
24 GAME

Masukkan Angka :
4 4 2 2
diproses...
4, 4, 2, dan 2

((2 * 4) + 4) * 2
2 * (4 + (4 * 2))
2 * (4 + (2 * 4))
((2 * 4) - 2) * 4
2 * ((4 * 2) + 4)
2 * ((2 * 4) + 4)
(4 + (4 * 2)) * 2
4 * ((4 * 2) - 2)
(4 + (2 * 4)) * 2
((4 * 2) + 4) * 2
4 * ((2 * 4) - 2)
((4 * 2) - 2) * 4

Hasil = 12 buah solusi
```

Kasus : 4 4 2 2

Hasil : 12 solusi

e.

```
PS C:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src> cd "c:\Users\IRGIANSYAH\OneDrive\Documents\Tucill_13521167\src\" ; if ($?) { g++ src.cpp -o src } ; if ($?) { .\src }
24 GAME

Masukkan Angka :
1 1 1 1
diproses...
1, 1, 1, dan 1

Hasil = 0 buah solusi
```

Kasus : 1 1 1 1

Hasil 0 solusi

f.

24 GAME

Masukkan Angka :

1 4 6 6

diproses...

1, 4, 6, dan 6

$6 + ((4 - 1) * 6)$

$(6 * (4 + 1)) - 6$

$(6 * (4 - 1)) + 6$

$6 - (6 * (1 - 4))$

$6 + (6 * (4 - 1))$

$6 - ((1 - 4) * 6)$

$(6 * (1 + 4)) - 6$

$((1 + 4) * 6) - 6$

$((4 + 1) * 6) - 6$

$((4 - 1) * 6) + 6$

Hasil = 10 buah solusi

Kasus : 1 4 6 6

Hasil : 0 solusi

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		✓