

Laporan Tugas Kecil Strategi Algoritma

Penyelesaian Permainan Kartu 24 dengan Algoritma Brute Force

K1/13521159/Sulthan Dzaky Alfaro

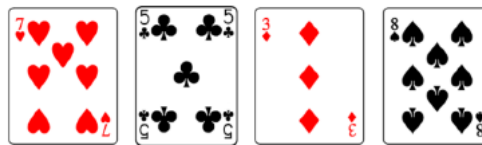
A. Algoritma Brute Force

Algoritma Brute Force merupakan cara penyelesaian suatu permasalahan dengan mengecek semua kemungkinan/kasus yang mungkin muncul. Algoritma ini termasuk algoritma yang fleksibel, maksudnya banyak permasalahan dapat diselesaikan dengan algoritma ini. Karena algoritma ini mengecek semua kemungkinan yang terjadi pada suatu permasalahan, sehingga waktu yang diperlukan algoritma untuk menyelesaikan permasalahan termasuk lama. Oleh karena itu algoritma ini tidak cukup efisien dalam menyelesaikan suatu permasalahan.

Dalam tugas kecil Strategi Algoritma ini, Penyelesaian Permainan Kartu 24 diselesaikan dengan menggunakan algoritma Brute Force. Algoritma yang saya gunakan terdiri atas beberapa tahap, diantaranya Membaca Input, Mencari Solusi, dan Menyimpan Hasil Solusi.

Membaca Input

Pada tahap pembacaan input, User harus menginputkan 4 angka atau huruf tertentu, antara lain A,2,3,4,5,6,7,8,9,10,J,Q,K. Tiap angka mewakili angka yang diinputkan dan untuk huruf merepresentasikan suatu angka, antara lain A bernilai 1, J bernilai 11, Q bernilai 12, dan K bernilai 13. Tiap angka atau huruf mewakili nilai dari tiap kartu yang nantinya akan dicari kombinasi dari 4 angka atau huruf tersebut dengan operasi dasar matematika (+,-,*,/) agar menghasilkan 24.



MAKE IT 24

Untuk pembacaan input, user memasukkan input dengan spasi sebagai pemisah (contoh: "A 5 K J"). Lalu program akan membaca dan mengkonversi huruf menjadi angka lalu tiap angka atau huruf (yang sudah dikonversikan) disimpan ke dalam array untuk proses selanjutnya.

Mencari Solusi

Untuk mencari solusi, disini menggunakan algoritma Brute Force. Pertama, saya membuat fungsi untuk menghitung setiap kasus. Kasus-kasus ini terdiri dari:

- ((a...b)...c)...d
- (a...(b...c))...d
- a...((b...c)...d)
- a...(b...(c...d))
- (a...b)...(c...d)

Ket: ... nanti diisi dengan sebuah operasi matematika(+,-,*,/)

Dan saya juga mengkonversi operasi-operasi matematika menjadi angka (+ angka 0, - angka 1, * angka 2, dan / angka 3) agar dalam mencari solusi menjadi lebih mudah karena dalam mencari solusi menggunakan algoritma Brute Force yang dimana akan menggunakan iterasi. Untuk mencari solusi, kita menggunakan iterasi untuk mencari semua kombinasi penempatan angka dan penempatan operasi matematika di tiap kasus. Kombinasi ini menggunakan indeks array yang tadi untuk menyimpan angka dari input user. Program akan mencari kombinasi angka serta operasi agar menghasilkan nilai 24. Apabila kombinasi tersebut menghasilkan 24, program akan menyimpan kombinasi tersebut kedalam set. Contohnya misalkan user menginputkan "A 5 9 K", program pertama akan mengkonversi menjadi angka semua dan memasukkannya ke dalam array, dengan isi array [1,5,9,13]. Lalu program akan mencari kombinasi yang dapat menghasilkan 24 dengan menggunakan iterasi. Contohnya kombinasi (1,0,5,2,13,1,9) yang mana memiliki arti $1+5*13-9$ yang nanti akan dimasukkan ke dalam fungsi untuk menghitung apakah hasil tersebut menghasilkan nilai 24 atau tidak di setiap kasus. Untuk contoh ini masuk ke dalam kasus 5 yang mana (a...b)...(c...d) berarti $(1+5)*(13-9)$ yang mana akan menghasilkan 24. Dan seterusnya program akan melakukan iterasi untuk mencari setiap kemungkinan yang ada.

Menyimpan Solusi

Solusi yang tadi sudah ditampilkan nantinya kita dapat menyimpan solusi yang tadi ditampilkan ke dalam file teks. Pertama kita akan ditanya apakah ingin menyimpan solusi kedalam file teks. Apabila ingin menyimpan, kita akan diminta untuk mengisi nama file untuk menyimpan solusi tersebut. Lalu program akan menyimpan semua solusi ke dalam file. Apabila tidak ada solusi, file akan berisi "Tidak ada solusi yang tersedia".

B. Source Code Program

Program terdiri dari 1 file yaitu *24Solver.cpp*. Didalam file tersebut terdapat fungsi *isSama*, *hitung_hitung*, dan *main* program.

isSama

```
bool isSama (double hasil)
{
    if(hasil-24<=0.00000001 && 24-hasil<=0.00000001)
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

Hitung_hitung

```
void hitung_hitung(double a,int oprt1,double b,int oprt2,double c,int oprt3,double d,set<string> * nyimpenstr,ofstream * datafile)
{
    //operator di inisialkan sebagai angka
    //'+' adalah 0
    //'-' adalah 1
    //'*' adalah 2
    //'/' adalah 3
    string oper1,oper2,oper3;
    if(oprt1==0)
    {
        oper1 = '+';
    }
    else if(oprt1==1)
    {
        oper1 = '-';
    }
    else if(oprt1==2)
    {
        oper1 = '*';
    }else if(oprt1==3)
    {
        oper1 = '/';
    }
    if(oprt2==0)
    {
        oper2 = '+';
    }
}
```

```
else if(oprt2==1)
{
    oper2 = '-';
}
else if(oprt2==2)
{
    oper2 = '*';
}else if(oprt2==3)
{
    oper2 = '/';
}
if(oprt3 ==0)
{
    oper3 = '+';
}
else if(oprt3==1)
{
    oper3 = '-';
}
else if(oprt3==2)
{
    oper3 = '*';
}else if(oprt3==3)
{
    oper3 = '/';
}
}
```

```

//-----
//Kasus 1 ((a..b)..c)..d
//-----
double kasus1;
double kasus12;
if(oprt1==0)
{
    kasus12=a+b;
}
else if(oprt1==1)
{
    kasus12=a-b;
}
else if(oprt1==2)
{
    kasus12=a*b;
}
else if(oprt1==3)
{
    kasus12=a/b;
}
double kasus13;
if(oprt2==0)
{
    kasus13=kasus12+c;
}

```

```

else if(oprt2==1)
{
    kasus13=kasus12-c;
}
else if(oprt2==2)
{
    kasus13=kasus12*c;
}
else if(oprt2==3)
{
    kasus13=kasus12/c;
}

double kasus14;
if(oprt3==0)
{
    kasus14=kasus13+d;
}
else if(oprt3==1)
{
    kasus14=kasus13-d;
}
else if(oprt3==2)
{
    kasus14=kasus13*d;
}

```

```

else if(oprt3==3)
{
    kasus14=kasus13/d;
}
kasus1 = kasus14;// hasil akhir kasus 1

```

```

//-----
//Kasus 2 (a..(b..c))..d
//-----

double kasus2;
double kasus21;
if(oprt2==0)
{
    kasus21 = b+c;
}
else if(oprt2==1)
{
    kasus21 = b-c;
}
else if(oprt2==2)
{
    kasus21 = b*c;
}
else if(oprt2==3)
{
    kasus21 = b/c;
}

```

```

double kasus22;
if(oprt1==0)
{
    kasus22 = a+kasus21;
}
else if(oprt1 == 1)
{
    kasus22 = a-kasus21;
}
else if(oprt1==2)
{
    kasus22 = a*kasus21;
}
else if(oprt1==3)
{
    kasus22 = a/kasus21;
}

```

```

double kasus23;
if(oprt3==0)
{
    kasus23 = kasus22+d;
}
else if(oprt3 == 1)
{
    kasus23 = kasus22-d;
}
else if(oprt3==2)
{
    kasus23 = kasus22*d;
}
else if(oprt3==3)
{
    kasus23 = kasus22/d;
}
kasus2 = kasus23;//hasil akhir kasus 2

```

```

//-----
//Kasus 3 a..((b..c)..d)
//-----
double kasus3;
double kasus31;
if(oprt2==0)
{
    kasus31 = b+c;
}
else if(oprt2==1)
{
    kasus31 = b-c;
}
else if(oprt2==2)
{
    kasus31 = b*c;
}
else if(oprt2==3)
{
    kasus31 = b/c;
}

```

```

double kasus32;
if(oprt3 ==0)
{
    kasus32 = kasus31+d;
}
else if(oprt3 ==1)
{
    kasus32 = kasus31-d;
}
else if(oprt3 ==2)
{
    kasus32 = kasus31*d;
}
else if(oprt3 ==3)
{
    kasus32 = kasus31/d;
}

```

```

double kasus33;
if(oprt1 == 0)
{
    kasus33 = a+kasus32;
}
else if(oprt1 == 1)
{
    kasus33 = a-kasus32;
}
else if(oprt1 == 2)
{
    kasus33 = a*kasus32;
}
else if(oprt1 == 3)
{
    kasus33 = a/kasus32;
}
kasus3 = kasus33; //hasil akhir kasus 3

```

```

//-----
//Kasus 4 a..(b..(c..d))
//-----
double kasus4;
double kasus41;
if(oprt3==0)
{
    kasus41 = c+d;
}
else if(oprt3==1)
{
    kasus41 = c-d;
}
else if(oprt3==2)
{
    kasus41 = c*d;
}
else if(oprt3==3)
{
    kasus41 = c/d;
}

```

```

double kasus44;
if(oprt2==0)
{
    kasus44 = b+kasus41;
}
else if(oprt2==1)
{
    kasus44 = b-kasus41;
}
else if(oprt2==2)
{
    kasus44 = b*kasus41;
}
else if(oprt2==3)
{
    kasus44 = b/kasus41;
}

```

```

double kasus43;
if(oprt1==0)
{
    kasus43 = a+kasus44;
}
else if(oprt1==1)
{
    kasus43 = a-kasus44;
}
else if(oprt1==2)
{
    kasus43 = a*kasus44;
}
else if(oprt1==3)
{
    kasus43 = a/kasus44;
}
kasus4 = kasus43;//Hasil akhir kasus 4

```

```

//-----
//Kasus 5 (a..b)..(c..d)
//-----
double kasus5;
double kasus51;
if(oprt1==0)
{
    kasus51 = a+b;
}
else if(oprt1==1)
{
    kasus51 = a-b;
}
else if(oprt1==2)
{
    kasus51 = a*b;
}
else if(oprt1==3)
{
    kasus51 = a/b;
}

```

```

double kasus52;
if(oprt3 == 0)
{
    kasus52 = c+d;
}
else if(oprt3==1)
{
    kasus52 = c-d;
}
else if(oprt3==2)
{
    kasus52 = c*d;
}
else if(oprt3==3)
{
    kasus52 = c/d;
}

```



```
double kasus53;  
if(oprt2 == 0)  
{  
    kasus53 = kasus51+kasus52;  
}  
else if(oprt2 == 1)  
{  
    kasus53 = kasus51-kasus52;  
}  
else if(oprt2 == 2)  
{  
    kasus53 = kasus51*kasus52;  
}  
else if(oprt2 == 3)  
{  
    kasus53 = kasus51/kasus52;  
}  
kasus5 = kasus53;
```

```
int dbtoint1 = int(a);  
int dbtoint2 = int(b);  
int dbtoint3 = int(c);  
int dbtoint4 = int(d);  
string strks1;  
string strks2;  
string strks3;  
string strks4;  
string strks5;  
string angka1 = to_string(dbtoint1);  
string angka2 = to_string(dbtoint2);  
string angka3 = to_string(dbtoint3);  
string angka4 = to_string(dbtoint4);  
string kss1 = to_string(kasus1);  
string kss2 = to_string(kasus2);  
string kss3 = to_string(kasus3);  
string kss4 = to_string(kasus4);  
string kss5 = to_string(kasus5);
```

```

//Print hasil

if(isSama(kasus1))
{
    strks1 = "("+angka1+oper1+angka2+"")+oper2+angka3+"")+oper3+angka4; //Kasus 1
    nyimpenstr->insert(strks1);
}
if(isSama(kasus2))
{
    strks2 = "("+angka1+oper1+"("+angka2+oper2+angka3+"))"+oper3+angka4; //Kasus 2
    nyimpenstr->insert(strks2);
}
if(isSama(kasus3))
{
    strks3 = angka1+oper1+"(("+angka2+oper2+angka3+"")+oper3+angka4+""); //Kasus 3
    nyimpenstr->insert(strks3);
}
if(isSama(kasus4))
{
    strks4 = angka1+oper1+"("+angka2+oper2+"("+angka3+oper3+angka4+"))"); //Kasus 4
    nyimpenstr->insert(strks4);
}

if(isSama(kasus5))
{
    strks5 = "("+angka1+oper1+angka2+"")+oper2+"("+angka3+oper3+angka4+""); //Kasus 5
    nyimpenstr->insert(strks5);
}

```

Main

```

int main()
{
    cout<<"-----"<<endl;
    cout<<"Selamat Datang di Game 24 Solver"<<endl;
    cout<<"-----"<<endl;
    int i;
    bool benarangka = true;
    bool benar = true;
    int a[4]; //buat menyimpan angka
    cout<<"Pilih cara mencari solusi(1/2)"<<endl;
    cout<<"1. Angka random"<<endl;
    cout<<"2. Memasukkan input"<<endl;
    cout<<"Pilih:";
    int pilihpilih;
    cin>>pilihpilih;
    if(pilihpilih==2)
    {
        char cari[15];
        cout<<"Pilih 4 angka atau huruf diantara angka atau huruf berikut:"<<endl;
        cout<<"A(dalam game ini bernilai 1)"<<endl<<"Angka 2 - 10"<<endl<<"J(dalam game ini bernilai 11)"<<endl;
        cout<<"Q(dalam game ini bernilai 12)"<<endl<<"K(dalam game ini bernilai 13)"<<endl;
        cout<<"Masukkan 4 jenis kartu. ";
        bool benarangka = true;
        bool benar = true;
        cin.getline(cari,15);
        int count=0;
        int panjang = strlen(cari);
    }
}

```

```

for(i=0;i<panjang;i++)
{
    if(cari[i]=='A')
    {
        a[count] = 1;
        count+=1;
    }
    else if (cari[i]=='2')
    {
        a[count] = 2;
        count+=1;
    }
    else if (cari[i]=='3')
    {
        a[count] = 3;
        count+=1;
    }
    else if (cari[i]=='4')
    {
        a[count] = 4;
        count+=1;
    }
    else if (cari[i]=='5')
    {
        a[count] = 5;
        count+=1;
    }
}

```

```

else if (cari[i]=='6')
{
    a[count] = 6;
    count+=1;
}
else if (cari[i]=='7')
{
    a[count] = 7;
    count+=1;
}
else if (cari[i]=='8')
{
    a[count] = 8;
    count+=1;
}
else if (cari[i]=='9')
{
    a[count] = 9;
    count+=1;
}
else if (cari[i]=='1' && cari[i+1]=='0')
{
    a[count] = 10;
    count+=1;
}
else if (cari[i]=='0' && cari[i-1]=='1')
{
    continue;
}

```

```

else if (cari[i]=='J')
{
    a[count] = 11;
    count+=1;
}
else if( cari[i]=='Q')
{
    a[count] = 12;
    count+=1;
}
else if(cari[i]=='K')
{
    a[count] = 13;
    count+=1;
}
else if(cari[i]==' ' && cari[i-1]!=' ' && cari[i+1]!=' ' )
{
    continue;
}
else if(cari[i]==' ' && cari[i-1]==' ' || cari[i+1]==' ' )
{
    benar = false;
}
else
{
    benar = false;
}
}

```

```

if(count==4 && panjang==7 || panjang==8)
{
    benarangka = true;
}
else
{
    benarangka = false;
}

```

```
while(!benar || !benarangka)
{
count=0;
cout<<"Tolong masukkan input yang benar"<<endl;
cout<<"Contoh input: 7 8 9 2"<<endl;
cout<<"Masukkan 4 jenis kartu: ";
cin.getline(cari,15);
benar = true;
int panjang = strlen(cari);
for(i=0;i<panjang;i++)
{
    if(cari[i]=='A')
    {
        a[count] = 1;
        count+=1;
    }
    else if (cari[i]=='2')
    {
        a[count] = 2;
        count+=1;
    }
    else if (cari[i]=='3')
    {
        a[count] = 3;
        count+=1;
    }
}
```

```
else if (cari[i]=='4')
{
    a[count] = 4;
    count+=1;
}
else if (cari[i]=='5')
{
    a[count] = 5;
    count+=1;
}
else if (cari[i]=='6')
{
    a[count] = 6;
    count+=1;
}
else if (cari[i]=='7')
{
    a[count] = 7;
    count+=1;
}
else if (cari[i]=='8')
{
    a[count] = 8;
    count+=1;
}
else if (cari[i]=='9')
{
    a[count] = 9;
    count+=1;
}
```

```
else if (cari[i]=='1' && cari[i+1]=='0')
{
    a[count] = 10;
    count+=1;
}
else if (cari[i]=='0' && cari[i-1]=='1')
{
    continue;
}
else if (cari[i]=='J')
{
    a[count] = 11;
    count+=1;
}
else if( cari[i]=='Q')
{
    a[count] = 12;
    count+=1;
}
else if(cari[i]=='K')
{
    a[count] = 13;
    count+=1;
}
else if(cari[i]==' ' && cari[i-1]!=' ' && cari[i+1]!=' ' )
{
    continue;
}
```

```

        else if(cari[i]==' ' && cari[i-1]!=' ' && cari[i+1]!=' ' )
        {
            continue;
        }
        else if(cari[i]==' ' && cari[i-1]==' ' || cari[i+1]==' ' )
        {
            benar = false;
        }
        else
        {
            benar = false;
        }
    }
    if(count==4 && panjang==7||panjang==8)
    {
        benarangka = true;
    }
    else
    {
        benarangka = false;
    }
}

```

```

else if(pilihpilih==1)
{
    cout<<"Angka yang didapat:"<<endl;
    cout<<endl;
    int i = 0;
    srand(time(NULL));
    for(i;i<4;i++)
    {
        a[i] = 1+(rand()%13);
        cout<<a[i]<<" ";
    }
}

```



```

cout<<endl;
    set<string> buatnyimpenstr;
    ofstream buang;
    clock_t start, end;
    start=clock();
    int jml=0;
    int x;
    int y;
    int z;
    int j;
    int k;
    int l;
    int m;

```

```

for (j=0;j<4;j++)
{
    for (k=0;k<4;k++)
    {
        for (l=0;l<4;l++)
        {
            for (m=0;m<4;m++)
            {
                for (x=0;x<4;x++)
                {
                    for (y=0;y<4;y++)
                    {
                        for (z=0;z<4;z++)
                        {
                            if(j!=k && j!=l && j!=m && k!=l && k!=m && l!=m)
                            {
                                hitung_hitung(a[j],x,a[k],y,a[l],z,a[m],&buatnyimpenstr,&buang);
                            }
                        }
                    }
                }
            }
        }
    }
}
}
buang.close();
jml = buatnyimpenstr.size();

```

```

if(jml==0)
{
    cout<<"Tidak ada solusi"<<endl;
}
else
{
    cout<<"Jumlah solusi="<<jml<<endl;
    for (auto& it : buatnyimpenstr) {
        cout << it <<endl;
    }
}
end = clock();
double waktu = double(end-start)/double(CLOCKS_PER_SEC);
cout<<"Time execution="<<fixed<<waktu<<" sec"<<endl;
cout<<"Apakah anda ingin memasukkan solusi ke dalam file?(y/n)=";
char nanya;
cin>>nanya;

```

```

if(nanya=='y')
{
    cout<<"Masukkan nama file untuk menyimpan solusi(.txt)="<<endl;
    string namafile;
    cin>>namafile;
    ofstream simpanfile(namafile);
    simpanfile<<"Kartu yang dipilih sebagai berikut:"<<endl;
    simpanfile<<a[0]<<" "<<a[1]<<" "<<a[2]<<" "<<a[3]<<endl;
    if(jml==0)
    {
        simpanfile<<"Tidak ada solusi yang tersedia"<<endl;
    }
    else
    {
        simpanfile<<"Berikut "<<jml<<" solusi yang tersedia:"<<endl;
        for (auto& it : buatnyimpenstr) {
            simpanfile << it <<endl;
        }
        simpanfile.close();
        cout<<"File berhasil disimpan. Game telah berakhir";
    }
}
else if(nanya=='n')
{
    cout<<endl<<"Oke, hasil solusi tidak disimpan. Game telah berakhir";
}

```

C. Screenshot Input and Output

1. 4 5 7 K

```
Masukkan 4 jenis kartu: 4 5 7 K
Jumlah solusi=8
((13*7)+5)/4
((7*13)+5)/4
(13-5)*(7-4)
(4-7)*(5-13)
(5+(13*7))/4
(5+(7*13))/4
(5-13)*(4-7)
(7-4)*(13-5)
Time execution=0.016000 sec
```

2. Q J 7 8

```
Masukkan 4 jenis kartu: Q J 7 8
Jumlah solusi=98
((11+12)+8)-7
((11+12)-7)+8
((11+8)+12)-7
((11+8)-7)+12
((11-7)+12)+8
((11-7)+8)+12
((12+11)+8)-7
((12+11)-7)+8
((12+8)+11)-7
((12+8)-7)+11
((12-7)+11)+8
((12-7)+8)+11
((8+11)+12)-7
((8+11)-7)+12
((8+12)+11)-7
((8+12)-7)+11
((8-7)+11)+12
((8-7)+12)+11
((11+12+8))-7
((11+(12-7))+8
((11+(8+12))-7
((11+(8-7))+12
((11+12)+(8-7)
((11+12)-(7-8)
((11+8)+(12-7)
((11+8)-(7-12)
((11-(7-12))+8
((11-(7-8))+12
((11-7)+(12+8)
((11-7)+(8+12)
((12*8)/(11-7)
((12+(11+8))-7
((12+(11-7))+8
((12+(8+11))-7
((12+(8-7))+11
((12+11)+(8-7)
((12+11)-(7-8)
((12+8)+(11-7)
((12+8)-(7-11)
((12-(7-11))+8
((12-(7-8))+11
((12-7)+(11+8)
((12-7)+(8+11)
((12/(11-7))*8
((8*12)/(11-7)
((8+(11+12))-7
((8+(11-7))+12
((8+(12+11))-7
((8+(12-7))+11
((8+11)+(12-7)
((8+11)-(7-12)
((8+12)+(11-7)
((8+12)-(7-11)
((8-(7-11))-12
((8-(7-12))-11
((8-(7-(11+12)))
((8-(7-(12+11)))
((8/(11-7))/12)
Time execution=0.015000 sec
```

3. K 2 9 8

Masukkan 4 jenis kartu: K 2 9 8

Jumlah solusi=21

$((13+8)-9)*2$

$((13-9)+8)*2$

$((8+13)-9)*2$

$((8-9)+13)*2$

$(13+(8-9))*2$

$(13-(9-8))*2$

$(13-9)*(8-2)$

$(2-8)*(9-13)$

$(8+(13-9))*2$

$(8-(9-13))*2$

$(8-2)*(13-9)$

$(9-13)*(2-8)$

$2*((13+8)-9)$

$2*((13-9)+8)$

$2*((8+13)-9)$

$2*((8-9)+13)$

$2*(13+(8-9))$

$2*(13-(9-8))$

$2*(8+(13-9))$

$2*(8-(9-13))$

$9/(2-(13/8))$

Time execution=0.016000 sec

4. 6 9 K Q

```
Jumlah solusi=32
((12/6)+13)+9
((12/6)+9)+13
((6+9)-13)*12
((6-13)+9)*12
((9+6)-13)*12
((9-13)+6)*12
(12-6)*(13-9)
(12/6)+(13+9)
(12/6)+(9+13)
(13+(12/6))+9
(13+9)+(12/6)
(13-9)*(12-6)
(6+(9-13))*12
(6-(13-9))*12
(6-12)*(9-13)
(9+(12/6))+13
(9+(6-13))*12
(9+13)+(12/6)
(9-(13-6))*12
(9-13)*(6-12)
12*((6+9)-13)
12*((6-13)+9)
12*((9+6)-13)
12*((9-13)+6)
12*(6+(9-13))
12*(6-(13-9))
12*(9+(6-13))
12*(9-(13-6))
13+((12/6)+9)
13+(9+(12/6))
```

```
9+((12/6)+13)
9+(13+(12/6))
Time execution=0.015000 sec
```

5. K Q 9 7

```
Masukkan 4 jenis kartu: K Q 9 7

Tidak ada solusi
Time execution=0.000000 sec
```

6. A 9 8 10

```
Masukkan 4 jenis kartu: A 9 8 10  
Tidak ada solusi  
Time execution=0.000000 sec
```

D. Extra

Link Github: https://github.com/SulthanDA28/Tucil1_13521159.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		