

Laporan Tugas Persegi (Rayhan Narindran 5024211022)

Source Code:

```
#include <iostream>
#include <vector>
#include <cmath>

using namespace std;

void showPP();
void addPP();
void plusPP(float xmin1, float xmin2, float xmax1, float xmax2, float
ymin1, float ymin2, float ymax1, float ymax2, int idx1, int idx2);
void minPP(float xmin1, float xmin2, float xmax1, float xmax2, float
ymin1, float ymin2, float ymax1, float ymax2, int idx1, int idx2);
void doublePP(float xmin, float xmax, float ymin, float ymax, float
idx);
void halfPP(float xmin, float xmax, float ymin, float ymax, float idx);
bool isIrisan(float& x11, float& x21, float& x12, float& x22, float&
y11, float& y21, float& y12, float& y22);

int first, second, fidx, sidx;

class pp{
public:
    float xmin, xmax, ymin, ymax, xtt, ytt, panjang, lebar, luas,
data[4];

    pp(float XMIN, float XMAX, float YMIN, float YMAX)
    {
        xmin = XMIN;
        xmax = XMAX;
        ymin = YMIN;
        ymax = YMAX;
        xtt = (XMAX - XMIN)/2;
        ytt = (YMAX - YMIN)/2;
        panjang = fabs(XMAX-XMIN);
        lebar = fabs(YMAX-YMIN);
        luas = panjang*lebar;
        data[0] = XMIN;
        data[1] = XMAX;
        data[2] = YMIN;
        data[3] = YMAX;
    }

    void operator+ (pp& pp)
    {
```

```

        plusPP(xmin, pp.xmin, xmax, pp.xmax, ymin, pp.ymin, ymax,
pp.ymax, fidx, sidx);
    };

    void operator- (pp& pp)
    {
        minPP(xmin, pp.xmin, xmax, pp.xmax, ymin, pp.ymin, ymax,
pp.ymax, fidx, sidx);
    };

    void operator++ ()
    {
        doublePP(xmin, xmax, ymin, ymax, fidx);
    };

    void operator-- ()
    {
        halfPP(xmin, xmax, ymin, ymax, fidx);
    };

    float operator[] (int pos)
    {
        return data[pos];
    }

    bool operator== (pp& pp)
    {
        return isIrisan(xmin, xmax, pp.xmin, pp.xmax, ymin, ymax,
pp.ymin, pp.ymax);
    };
};

vector<pp> vecPP;

int main()
{
    int choice;

    menu:
    cout << "PERSEGI PANJANG MENU:" << endl;
    cout << "1. Tampilkan semua persegi panjang" << endl;
    cout << "2. Tambah Persegi Panjang Baru" << endl;
    cout << "3. Penggabungan dua persegi panjang (harus beririsan)" <<
endl;
    cout << "4. Irisan dua persegi panjang" << endl;
    cout << "5. Kali 2 luasan persegi panjang" << endl;
    cout << "6. Bagi 2 luasan persegi panjang" << endl;
    cout << "7. Tunjukkan data persegi spesifik" << endl;

```

```

cout << "8. Cek irisan 2 persegi" << endl;
cout << "0. Tutup program" << endl;
cout << "Pilihan: ";
cin >> choice;
cout << endl;

switch(choice){
    case 0:
        return 0;
        break;

    case 1:
        if (vecPP.empty() == true) {
            cout << "Tidak ada persegi panjang!" << endl;
        }
        else{showPP();}
        cout << endl;
        goto menu;
        break;

    case 2:
        addPP();
        cout << endl;
        goto menu;
        break;

    case 3:
        if (vecPP.size() < 2)
        {
            cout << "TERSEDIA KURANG DARI 2 PERSEGI!" << endl;
        }
        else
        {
            cout << "Persegi pertama: ke-";
            cin >> first;
            cout << "Persegi kedua: ke-";
            cin >> second;

            fidx = first - 1;
            sidx = second - 1;

            if (vecPP[fidx] == vecPP[sidx]){
                vecPP[fidx] + vecPP[sidx];
            } else {
                cout << endl;
                cout << "PERSEGI PANJANG TIDAK BERIRISAN" << endl;
            };
        }
}

```

```

        cout << endl;
        goto menu;
        break;

case 4:
    if (vecPP.size() < 2)
    {
        cout << "TERSEDIA KURANG DARI 2 PERSEGI!" << endl;
    }
    else
    {
        cout << "Persegi pertama: ke-";
        cin >> first;
        cout << "Persegi kedua: ke-";
        cin >> second;

        fidx = first - 1;
        sidx = second - 1;

        if (vecPP[fidx] == vecPP[sidx]){
            vecPP[fidx] - vecPP[sidx];
        } else {
            cout << endl;
            cout << "PERSEGI PANJANG TIDAK BERIRISAN" << endl;
        }
    }

    cout << endl;
    goto menu;
    break;

case 5:
    cout << "Persegi ke-: ";
    cin >> first;

    fidx = first - 1;

    ++vecPP[fidx];

    cout << endl;
    goto menu;
    break;

case 6:
    cout << "Persegi ke-: ";
    cin >> first;

```

```

        fidx = first - 1;

        --vecPP[fidx];

        cout << endl;
        goto menu;
        break;

    case 7:
        cout << "Persegi ke-: ";
        cin >> first;

        fidx = first - 1;

        cout << "XMIN: " << vecPP[fidx][0] << endl;
        cout << "XMAX: " << vecPP[fidx][1] << endl;
        cout << "YMIN: " << vecPP[fidx][2] << endl;
        cout << "YMAX: " << vecPP[fidx][3] << endl;

        cout << endl;
        goto menu;
        break;

    case 8:
        if (vecPP.size() < 2)
        {
            cout << "TERSEDIA KURANG DARI 2 PERSEGI!" << endl;
        }
        else
        {
            cout << "Persegi pertama: ke-";
            cin >> first;
            cout << "Persegi kedua: ke-";
            cin >> second;

            fidx = first - 1;
            sidx = second - 1;

            cout << endl;
            if (vecPP[fidx] == vecPP[sidx])
            {
                cout << "PERSEGI BERIRISAN" << endl;
            }
            else
            {
                cout << "PERSEGI TIDAK BERIRISAN" << endl;
            }
        }
    }
}

```

```

        cout << endl;
        goto menu;
        break;

    default:
        cout << "Invalid Input!" << endl;
        goto menu;
        break;
};

};

void showPP()
{
    cout <<
    "===== " <<
    endl;
    cout << "Semua Persegi Panjang: " << endl;
    for (long unsigned int i = 0; i < vecPP.size(); i++){
        cout << "Persegi ke-" << i+1 << ": " << endl;
        cout << "x-min      : " << vecPP[i].xmin << endl;
        cout << "x-max      : " << vecPP[i].xmax << endl;
        cout << "y-min      : " << vecPP[i].ymin << endl;
        cout << "y-max      : " << vecPP[i].ymax << endl;
        cout << "y-tengah   : " << vecPP[i].ytt << endl;
        cout << "x-tengah   : " << vecPP[i].xtt << endl;
        cout << "panjang    : " << vecPP[i].panjang << endl;
        cout << "lebar      : " << vecPP[i].lebar << endl;
        cout << "luas       : " << vecPP[i].luas << endl;
        cout << endl;
    }
    cout <<
    "===== " <<
    endl;
    cout << endl;
};

void addPP()
{
    float xmin, xmax, ymin, ymax;
    ppinput:
    cout << "PENAMBAHAN PERSEGI" << endl;
    cout << "Berapa x-min dari persegi: ";
    cin >> xmin;
    cout << "Berapa x-max dari persegi: ";
    cin >> xmax;
    cout << "Berapa y-min dari persegi: ";
    cin >> ymin;

```

```

    cout << "Berapa y-max dari persegi: ";
    cin >> ymax;

    if (xmax < xmin){swap(xmin, xmax);};
    if (ymax < ymin){swap(ymin, ymax);};
    if (xmax == xmin){cout << "xmax cant be equal to xmin!" << endl;
goto ppinput;};
    if (ymax == ymin){cout << "ymax cant be equal to ymin!" << endl;
goto ppinput;};

    pp newPP(xmin, xmax, ymin, ymax);
    vecPP.push_back(newPP);

    cout << "Persegi panjang berhasil ditambahkan!" << endl;
};

void plusPP(float xmin1, float xmin2, float xmax1, float xmax2, float
ymin1, float ymin2, float ymax1, float ymax2, int idx1, int idx2)
{
    float kiri, kanan, bawah, atas;

    if (xmin1 <= xmin2)
    {
        kiri = xmin1;
    }
    else if (xmin1 >= xmin2)
    {
        kiri = xmin2;
    }

    if (xmax1 <= xmax2)
    {
        kanan = xmax2;
    }
    else if (xmax1 >= xmax2)
    {
        kanan = xmax1;
    }

    if (ymin1 <= ymin2)
    {
        bawah = ymin1;
    }
    else if (ymin1 >= ymin2)
    {
        bawah = ymin2;
    }
}

```

```

        if (ymax1 <= ymax2)
        {
            atas = ymax2;
        }
        else if (ymax1 >= ymax2)
        {
            atas = ymax1;
        }

        pp replacePP(kiri, kanan, bawah, atas);
        vecPP[idx1] = replacePP;
        vecPP.erase(vecPP.begin()+idx2);

        cout << "PERSEGI BERHASIL DIGABUNGKAN!" << endl;
    }

void minPP(float xmin1, float xmin2, float xmax1, float xmax2, float
ymin1, float ymin2, float ymax1, float ymax2, int idx1, int idx2)
{
    float kiri, kanan, atas, bawah;

    if (xmin2 > xmin1 and xmin2 < xmax1)
    {
        kiri = xmin2;
        kanan = xmax1;
    }
    else if (xmin1 > xmin2 and xmin1 < xmax2)
    {
        kiri = xmin1;
        kanan = xmax2;
    }

    if (ymin2 > ymin1 and ymin2 < ymax1)
    {
        bawah = ymin2;
        atas = ymax1;
    }
    else if (ymin1 > ymin2 and ymin1 < ymax2)
    {
        bawah = ymin1;
        atas = ymax2;
    }

    pp replacePP(kiri, kanan, bawah, atas);
    vecPP[idx1] = replacePP;
    vecPP.erase(vecPP.begin()+idx2);

    cout << "IRISAN PERSEGI BERHASIL DIAMBIL!" << endl;
}

```



```

};

void doublePP(float xmin, float xmax, float ymin, float ymax, float
idx)
{
    xmax = ((xmax-xmin) * sqrt(2)) + xmin;
    ymax = ((ymax-ymin) * sqrt(2)) + ymin;

    pp replacePP(xmin, xmax, ymin, ymax);
    vecPP[idx] = replacePP;

    cout << "PERSEGI BERHASIL DIKALI 2 LUASNYA!" << endl;
};

void halfPP(float xmin, float xmax, float ymin, float ymax, float idx)
{
    xmax = ((xmax-xmin) / sqrt(2)) + xmin;
    ymax = ((ymax-ymin) / sqrt(2)) + ymin;

    pp replacePP(xmin, xmax, ymin, ymax);
    vecPP[idx] = replacePP;

    cout << "PERSEGI BERHASIL DIBAGI 2 LUASNYA!" << endl;
};

bool isIrisan(float& x11, float& x21, float& x12, float& x22, float&
y11, float& y21, float& y12, float& y22)
{
    bool xsinggung;
    bool ysinggung;

    if (((x11 < x12 and x11 > x22) or (x11 > x12 and x11 < x22)) or
((x21 < x12 and x21 > x22) or (x21 > x12 and x21 < x22)))
    {
        xsinggung = true;
    }
    else{xsinggung = false;};

    if (((y11 < y12 and y11 > y22) or (y11 > y12 and y11 < y22)) or
((y21 < y12 and y21 > y22) or (y21 > y12 and y21 < y22)))
    {
        ysinggung = true;
    }
    else{ysinggung = false;};

    if (xsinggung == true and ysinggung == true){
        return true;
    } else {return false;};
}

```

```
};
```

Pada program ini saya ditugaskan untuk menghasilkan sebuah class yang bisa memiliki objek sebuah persegi dengan atribut xmin, xmax, ymin, ymax. Disini saya menambahkan beberapa atribut seperti ytt yaitu y tengah dari persegi, xtt yaitu x tengah dari persegi, lebar dari persegi, panjang dari persegi, dan juga luas. Kemudian class tersebut juga melakukan operator overloading, dimana operator + digunakan untuk menggabungkan luas dari dua persegi menjadi satu dengan syarat kedua persegi tersebut beririsan. Operator – digunakan untuk mengambil irisan dari dua buah persegi, operator ++ digunakan untuk mengalikan 2 luas sebuah persegi dengan menambahkan panjang dan lebar dari persegi tersebut, operator – digunakan untuk membagi 2 luas sebuah persegi dengan cara yang sama, operator [] digunakan untuk memperlihatkan xmin, xmax, ymin, dan ymax dari sebuah persegi, dan operator == digunakan untuk melihat apakah dua buah persegi beririsan.

Pada program saya menamakan classnya PP, dan objek class tersebut akan disimpan dalam sebuah vector yang dinamakan vecPP, kemudian saya juga membuat menu untuk program ini sesuai dengan fungsi fungsi yang sudah di hasilkan. Penulis tidak menggunakan headerfile dikarenakan hanya menggunakan satu buah class, sehingga lebih efisien jika hanya diletakkan pada satu file.

Untuk algoritma yang digunakan dalam program ini, pertama untuk mencari jika 2 persegi beririsan, yang penulis lakukan adalah melihat apakah xmin dan xmax kedua buah persegi beririsan begitu juga dengan ymin dan ymax, jika keduanya beririsan maka persegi juga beririsan. Untuk menggabungkan 2 buah persegi, penulis mencari xmin dan ymin terkecil dari dua persegi, dan xmax dan ymax terbesar dari 2 buah persegi. Untuk mencari irisan penulis mencari titik titik perpotongan antara 2 persegi yang kemudian dijadikan persegi baru, untuk mengalikan 2 luas sebuah persegi, penulis mengalikan panjang juga lebar dari persegi dengan akar 2, kemudian dijadikan panjang dan lebar yang baru, untuk membagi 2 persegi digunakan cara yang sama, namun membagi panjang dan lebar dengan akar 2.