#include "stdafx.h"

#include "iostream"

#include "math.h"

using namespace std;

#define PI 3.14159265358979

class project {

private:

double x, y, v, a, t,g=10;

public:

project() {

cout << "##计算抛体运动位置##" << endl;

cout << "请输入初速度v(m/s):";

cin >> v;

cout << "请输入初速度方向a（°）：";

cin >> a;

a = a \* PI / 180;

cout << "请输入时刻t(s)：";

cin >> t;

x = v \* cos(a)\*t;

y = v \* sin(a)\*t - 0.5\*g\*t\*t;

cout << "此时物体的x坐标是" << x << endl;

cout << "此时物体的y坐标是" << y << endl;

}

};

class fire {

private:

double x, y, v, g = 10;

public:

fire() {

cout << "##计算开炮方向##" << endl;

cout << "请输入初速度v(m/s):";

cin >> v;

cout << "“二营长！你他娘的意大利炮呢？！”" << endl;

cout << "“团长，轰哪儿？”" << endl;

cout << "请输入目标的x坐标：";

cin >> x;

cout << "请输入目标的y坐标：";

cin >> y;

double a = g \* x\*x / (2 \* v\*v);

double delta = x\*x - 4 \* a\*(a + y);

if (delta < 0) cout << "无法打中目标" << endl;

else {

double m1 = (x + pow(delta, 0.5)) / (2\*a);

double m2 = (x - pow(delta, 0.5)) / (2 \* a);

cout << "开炮方向为" << atan(m1)\*180/PI << "度，或" << atan(m2)\*180/PI << "度。"<<endl;

cout << "“开炮！！！”" << endl;

}

}

};

class PEC {

private:

double m1, m2, v1, v2,temp;

public:

PEC() {

cout << "##计算完全弹性碰撞##" << endl;

cout << "请输入左侧物体的质量m1(kg):" ;

cin >> m1;

cout << "请输入左侧物体的速度v1(m/a):";

cin >> v1;

cout << "请输入右侧物体的质量m2(kg):";

cin >> m2;

cout << "请输入右侧物体的速度v2(m/a):";

cin >> v2;

cout << "BOOM!!!"<<endl;

temp = ((m1 - m2)\*v1 + 2 \* m2\*v2) / (m1 + m2);

cout << "碰撞后左侧物体的速度=" << temp << endl;

temp = (2 \* m1\*v1 + (m2 - m1)\*v2) / (m1 + m2);

cout << "碰撞后右侧物体的速度=" << temp << endl;

}

};

class lorentz {

private:

double λ, v, b, t, a;

public:

lorentz() {

cout << "##计算带电粒子在磁场中的运动" << endl;

cout << "请输入粒子荷质比（C/kg）:";

cin >> λ;

cout << "请输入粒子初速度大小（m/s):";

cin >> v;

cout << "请输入粒子初速度方向（°）:";

cin >> a;

cout << "请输入磁场大小（T）（以垂直入纸为正向）:";

cin >> b;

cout << "请输入时刻（s）:";

cin >> t;

double r = v / b / λ;

double x0 = r \* cos((a - 90)\*PI / 180);

double y0 = r \* sin((a - 90)\*PI / 180);

double d = v \* t / r;

double x = x0 + r \* cos((a + 90)\*PI / 180 + d);

double y = y0 + r \* sin((a + 90)\*PI / 180 + d);

cout << "此时的x坐标是" << x << endl;

cout << "此时的y坐标是" << y << endl;

}

};

class adiabatic {

private:

double p1, v1,p2,v2, i,w;

public:

adiabatic() {

cout << "##计算绝热过程做功##" << endl;

cout << "请选择气体类型" << endl;

cout << "A 单原子气体\t" << "B 双原子气体\t" << "C 多原子气体"<<endl;

char m;

cin >> m;

switch (m) {

case'A':i=3; break;

case'B':i=5; break;

case'C':i = 6; break;

default:cout << "错误输入" << endl;

}

double γ = i + 2 / i;

cout << "请输入初始压强（kPa):";

cin >> p1;

cout << "请输入初始体积（立方米）:";

cin >> v1;

cout << "请选择变化的物理量（A、压强 B、体积）：";

cin >> m;

switch (m) {

case'A': {

cout << "请输入变化后的压强（kPa):";

cin >> p2;

v2 = v1 \* pow((p1 / p2), 1 / γ);

cout << "变化后的体积=" << v2 << "m³" << endl;

}break;

case'B': {

cout << "请输入变化后的体积（立方米）:";

cin >> v2;

p2 = p1 \* pow((v1 / v2), γ);

cout << "变化后的压强=" << p2 << "kPa" << endl;

} break;

default:cout << "错误输入" << endl;

}

w = i / 2 \* p1\*v1\*(1 - pow((v1 / v2), γ - 1));

cout << "该过程做的功=" << w << "kJ"<<endl;

}

};

class diffraction {

private:

double d, a, bochang;

int n, m;

public:

diffraction() {

cout << "##计算光栅衍射的强度和条纹##" << endl;

cout << "请输入光栅常数（nm)：";

cin >> d;

cout << "请输入缝宽（nm）:";

cin >> a;

cout << "请输入光的波长（nm）:";

cin >> bochang;

cout << "请输入级数：";

cin >> n;

if ((a\*n / d) - (int)(a\*n / d) == 0) cout << "缺级" << endl;

else {

cout << "请输入缝数：";

cin >> m;

double θ = asin(n\*bochang / d);

cout << "衍射角=" << θ \* 180 / PI << "°" << endl;

double v = PI \* n;

double u = PI \* a\*n / d;

double k = pow((sin(u) / u), 2)\*pow(sin(m\*v) / sin(v), 2);

cout << "（以入射光强为I0）光强大小=" << k << "I0" << endl;

}

}

};

void main() {

cout << "##欢迎使用物理计算助手##" << endl;

cout << "##请选择你想要使用的功能##" << endl;

cout << "0 结束使用" << endl;

cout << "1 计算抛体运动位置" << endl;

cout << "2 计算开炮方向" << endl;

cout << "3 计算完全弹性碰撞" << endl;

cout << "4 计算带电粒子在磁场中的运动" << endl;

cout << "5 计算绝热过程做功" << endl;

cout << "6 计算光栅衍射的强度和条纹" << endl;

int n = 99;

while (n != 0) {

cout << "请选择使用的功能: ";

cin >> n;

switch (n) {

case 0:cout << "欢迎下次使用" << endl; break;

case 1: { project Myproject; }break;

case 2: {fire Myfire; }break;

case 3: {PEC MyPEC; }break;

case 4: {lorentz Mylorentz; } break;

case 5: {adiabatic Myadiabatic; }break;

case 6: {diffraction Mydiffraction; }break;

default:cout << "错误输入" << endl;

}

}

};

