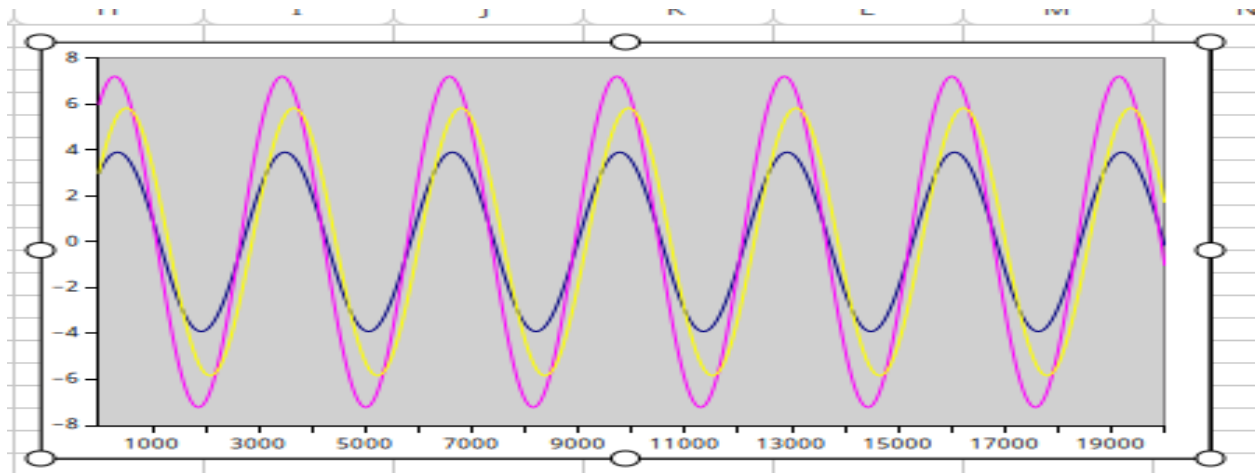


Anisotropic harmonic oscillator



```
#include<iostream>
#include<fstream>
#include<cmath>
using namespace std;
double f1(double x1,double x2,double t){
return x2;
}
double f2(double x1,double omega_x,double t){
return -omega_x*omega_x*x1;
}
double f3(double x3,double x4,double t){
return x4;
}
double f4(double x3,double omega_y,double t){
return -omega_y*omega_y*x3;
}
double f5(double x5,double x6,double t){
return x6;
}
double f6(double x5,double omega_z,double t){
return -omega_z*omega_z*x5;
}
int main(){
double T=20.0;
double tau=0.001;
int nmax=T/tau;
double x1[nmax+1],x2[nmax+1];
double x3[nmax+1],x4[nmax+1];
double x5[nmax+1],x6[nmax+1];
cout<<"x1[0]:";cin>>x1[0];
cout<<"x2[0]:";cin>>x2[0];
cout<<"x3[0]:";cin>>x3[0];
cout<<"x4[0]:";cin>>x4[0];
```

```

cout<<"x5[0]:";cin>>x5[0];
cout<<"x6[0]:";cin>>x6[0];
double omega_x,omega_y,omega_z;
cout<<"omega_x:";cin>>omega_x;
cout<<"omega_y:";cin>>omega_y;
cout<<"omega_z:";cin>>omega_z;
ofstream output("oscillator.txt");
for(int n=0;n<nmax;n++){
double t=n*tau;
double k1[6],k2[6],k3[6],k4[6];
k1[0]=f1(x1[n],x2[n],t);
k1[1]=f2(x1[n],omega_x,t);
k1[2]=f3(x3[n],x4[n],t);
k1[3]=f4(x3[n],omega_y,t);
k1[4]=f5(x5[n],x6[n],t);
k1[5]=f6(x5[n],omega_z,t);

k2[0]=f1(x1[n]+0.5*tau*k1[0],x2[n]+0.5*tau*k1[1],t+0.5*tau);
k2[1]=f2(x1[n]+0.5*tau*k1[0],omega_x,t+0.5*tau);
k2[2]=f3(x3[n]+0.5*tau*k1[2],x4[n]+0.5*tau*k1[3],t+0.5*tau);
k2[3]=f4(x3[n]+0.5*tau*k1[2],omega_y,t+0.5*tau);
k2[4]=f5(x5[n]+0.5*tau*k1[4],x6[n]+0.5*tau*k1[5],t+0.5*tau);
k2[5]=f6(x5[n]+0.5*tau*k1[4],omega_z,t+0.5*tau);

k3[0]=f1(x1[n]+0.5*tau*k2[0],x2[n]+0.5*tau*k2[1],t+0.5*tau);
k3[1]=f2(x1[n]+0.5*tau*k2[0],omega_x,t+0.5*tau);
k3[2]=f3(x3[n]+0.5*tau*k2[2],x4[n]+0.5*tau*k2[3],t+0.5*tau);
k3[3]=f4(x3[n]+0.5*tau*k2[2],omega_y,t+0.5*tau);
k3[4]=f5(x5[n]+0.5*tau*k2[4],x6[n]+0.5*tau*k2[5],t+0.5*tau);
k3[5]=f6(x5[n]+0.5*tau*k2[4],omega_z,t+0.5*tau);

k4[0]=f1(x1[n]+tau*k3[0],x2[n]+tau*k3[1],t+tau);
k4[1]=f2(x1[n]+tau*k3[0],omega_x,t+tau);
k4[2]=f3(x3[n]+tau*k3[2],x4[n]+tau*k3[3],t+tau);
k4[3]=f4(x3[n]+tau*k3[2],omega_y,t+tau);
k4[4]=f5(x5[n]+tau*k3[4],x6[n]+tau*k3[5],t+tau);
k4[5]=f6(x5[n]+tau*k3[4],omega_z,t+tau);

x1[n+1]=x1[n]+tau*(k1[0]+2*k2[0]+2*k3[0]+k4[0])/6.0;
x2[n+1]=x2[n]+tau*(k1[1]+2*k2[1]+2*k3[1]+k4[1])/6.0;
x3[n+1]=x3[n]+tau*(k1[2]+2*k2[2]+2*k3[2]+k4[2])/6.0;
x4[n+1]=x4[n]+tau*(k1[3]+2*k2[3]+2*k3[3]+k4[3])/6.0;
x5[n+1]=x5[n]+tau*(k1[4]+2*k2[4]+2*k3[4]+k4[4])/6.0;
x6[n+1]=x6[n]+tau*(k1[5]+2*k2[5]+2*k3[5]+k4[5])/6.0;

output<<t<<"\t"<<x1[n]<<"\t"<<x2[n]<<"\t"<<x3[n]<<"\t"<<x4[n]<<"\t"<<x
5[n]<<"\t"<<x6[n]<<"\n";
}
output.close();
return 0;

```

