MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

Bir dosyaya ciktilari yazmak icin kullanılacak komut
int fprintf (FILE * stream, const char * format, ...);

specifier	Output	Example
d *or* i	Signed decimal integer	392
u	Unsigned decimal integer	7235
o	Unsigned octal	610
x	Unsigned hexadecimal integer	7fa
X	Unsigned hexadecimal integer (uppercase)	7FA
f	Decimal floating point, lowercase	392.65
F	Decimal floating point, uppercase	392.65
e	Scientific notation (mantissa/exponent), lowercase	3.9265e+2
E	Scientific notation (mantissa/exponent), uppercase	3.9265E+2
g	Use the shortest representation: %e or %f	392.65
G	Use the shortest representation: %E or %F	392.65
a	Hexadecimal floating point, lowercase	-0xc.90fep-2
A	Hexadecimal floating point, uppercase	-0XC.90FEP-2
c	Character	a
s	String of characters	sample
p	Pointer address	b8000000
n	Nothing printed. The corresponding argument must be a pointer to a signed int. The number of characters written so far is stored in the pointed location.	
%	A % followed by another % character will write a single % to the stream.	%

```
/* fprintf example */
#include <stdio.h>
#include <iostream>
using namespace std;
int main ()
{
   FILE * pFile;
   int n;
   char name [100];
   pFile = fopen ("myfile.txt","w");
   for (n=0 ; n<3 ; n++)
   {
     printf ("please, enter a name: ");
     cin>>name;
     fprintf (pFile, "Name %d [%-10.10s]\n",n+1,name);
   }
   fclose (pFile);
   return 0;
```

MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

Türev

```
/*Program 1.1 f(x)=cosx fonksiyonunun x=1 noktasýnda sayýsal türevi ve hata payý*/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

main()
{
	int i;
	double a, F1, h;
	a = 1.e0;
	h = 0.1e0;
	printf("	H		f'(1)		Gercek Deger-f'(1)\n");
	for(i = 1; i <= 14; i++){
		F1 = (sin(a + h) - sin(a))/h;
		printf("%-15.14lf  %-11.10lf  %-11.10lf\n", h, F1, (cos(a) - F1));
		h /= 10;
	}
	printf("turevin gercek degeri %11.10lf", cos(1));

	return 0;
}
```

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

Türev2

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
//#define F(x) ((double)tan(x))
//#define F1(x) ((double)(1 / pow(cos(x),2)))
main()
{
	int i;
	double f1_ileri, f1_geri, f1_simetrik, f1_tam, h, x;
		h = 0,001;
	printf("   x        f1_ileri        f1_geri        f1_simetrik        f1_tam \n");
	for(i = 1; i <= 3; i++){
		x = i;
		f1_ileri = (tan(x + h) - tan(x)) / h;
		f1_geri = (tan(x) - tan(x - h)) / h;
		f1_simetrik = (tan(x + h) - tan(x - h)) / (2 * h);
		f1_tam = 1 / pow(cos(x),2);
		printf("  %4.2lf %16.10lf %11.10lf %16.10lf %16.10lf \n", x, f1_ileri,
f1_geri, f1_simetrik, f1_tam);
	}

	return 0;
}
```

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

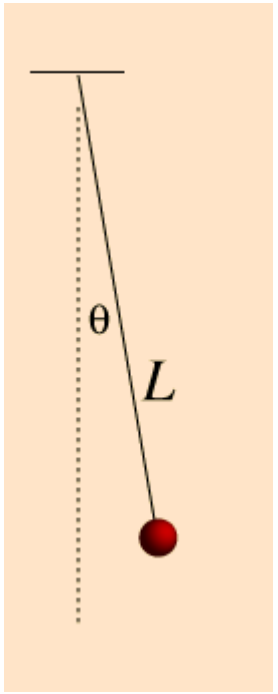
Türev3

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
main()
{
	int i;
	double f1_simetrik, f1_tam, fark, h, x;

	h = 0.1;
	x = 1.0;
	printf("    h                  f1_simetrik          f1_tam          fark\n");
	for(i = 1; i <= 12; i++){
		f1_simetrik = (exp(x + h) - exp(x - h)) / (2 * h);
		f1_tam = exp(x);
		fark = f1_tam - f1_simetrik;
		printf("  %16.12lf   %16.10lf   %16.10lf   %16.10lf\n", h,
f1_simetrik, f1_tam, fark);
		h /= 10.0;
	}

	return 0;
}
```

MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

Basit Sarkaç



$$\theta = \theta_{\max} \sin \sqrt{\frac{g}{L}} t$$

$$\omega = \sqrt{\frac{g}{L}}$$

$$f = \frac{1}{2\pi} \sqrt{\frac{g}{L}}$$

$$T = 2\pi \sqrt{\frac{L}{g}}$$

MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

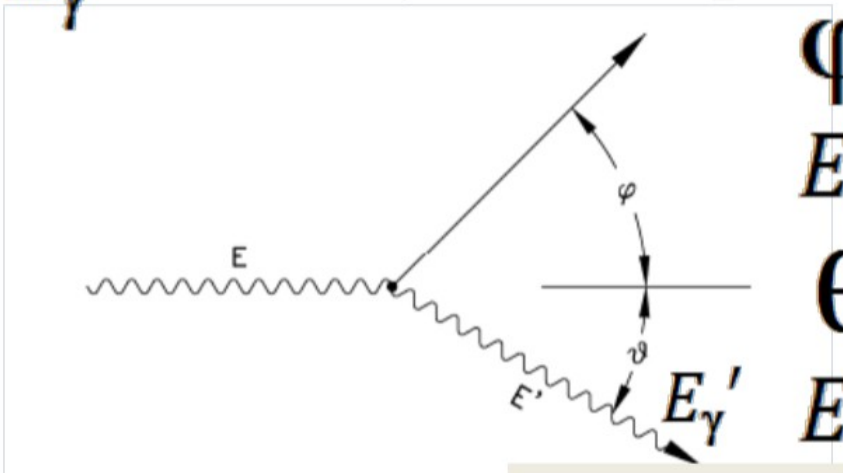
Sarkac.cpp

```
// Bu program Basit sarkacın periyodunu kullanarak
//yerçekimi ivmesni hesaplamaktadır

#include <iostream>
#include <cmath>
using namespace std;
float const PI = 3.141593;
int main() {
   float length, period;
   float g_base; // [m/s^2] baseline value of g
   cout << "This program evaluates the local value of g[m/s^2]\n"
        << "as a function of pendulum period\n";
   cout << "\nEnter the pendulum length [m]: ";
   cin >> length;
   cout << "enter period [s]: ";
   cin >> period;
   g_base = 9.807; // [m/s^2]
   cout << "pendulum length set at " << length << "[m]\n";
   cout << "for baseline gravity of " << g_base << "[m/s^2]\n";
   cout << "small angle period is "
          << 2*PI*sqrt(length/g_base) << "[s]\n";

   float g = 4*PI*PI*length/(period*period);
   cout << "for period = " << period << " seconds, "

        << "g = " << g << "[m/s^2]\n";
   return 0;
}
```

COMPTON SAÇILMASI

Bir fotonun serbest olduğu varsayılan bir elektron tarafından saçılması sürecidir.

E_γ : Gelen Fotonun Enerjisi



φ : Elektronun Saçılım Açısı

E_e :Saçılan Elektronun Enerjisi

θ : Fotonun Saçılım Açısı

E_γ' : Saçılan Fotonun Enerjisi

MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

$$E'_\gamma = f(\theta)E_\gamma \qquad f(\theta) = \frac{1}{1+\alpha(1-cos(\theta))} \qquad \alpha = \frac{E_\gamma}{m_e c^2}$$

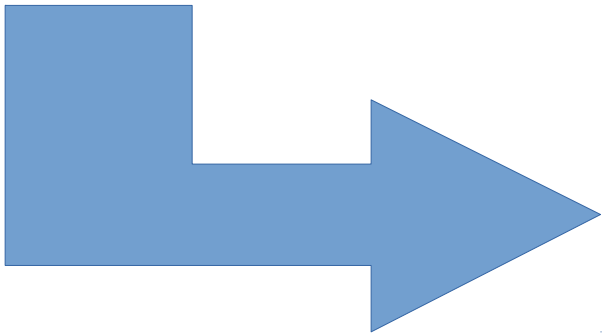
$$m_e c^2 = 0.511\,MeV$$

MİMAR SİNAN FINE ARTS UNIVERSITY

C++ Giriş Ders 5
MSGSU Fizik Bölümü
Ferhat ÖZOK

Compton.cpp

```
#include <stdio.h>
#include <iostream>
#include <math.h>
using namespace std;
const double mec2 = 0.511;//MeV
const double PI_T = 3.141592653589793;
const double PI   = 2.0*acos(0.0);
const double DegdenRad  = PI/180.0;
double Eg,EgU,Ec,teta;
//char ch;
FILE * ptr_Dosya;
int main(){
//////////////////////////////////////////////////////
cout <<" Gelen Foton Enerjisi(Eg[MeV]): ";cin >> Eg;
teta=180;
cout <<" Sacilma Acisi (teta[Derece]) : "<<teta<<endl;
```



```
//////////////////////////////////////////////////////
double FcosTETA = 1.0-cos(teta*DegdenRad);
//printf("%1.15f %1.15f %1.15",PI,PI_T);ch=getch();
double alfa = Eg/mec2;
EgU = Eg/(1.0+alfa*FcosTETA);
Ec  = Eg-EgU;
cout <<" Sacilan Foton Enerjisi(EgU[MeV]): "<<EgU<<endl;
cout <<" Ec[MeV] : "<<Ec<<endl;

//////////////////////////////////////////////////////
cout<<"------------------------------\n";
cout<<"cos(teta) teta      EgU  Eg-EgU\n";
cout<<"------------------------------\n";
double cosTETA = -1.0;
do{
   EgU = Eg/(1.0+alfa*(1-cosTETA));
   printf("  %+1.2f   %2.2f    %3.2f  %3.2f\n",
         cosTETA,acos(cosTETA)/DegdenRad,EgU,Eg-EgU);
   cosTETA += 0.01;
}while (cosTETA < 1.0);



/**/
ptr_Dosya = fopen("Ders6-01.dat","w");
cosTETA = -1.0;
do{
   EgU = Eg/(1.0+alfa*(1-cosTETA));
   fprintf(ptr_Dosya,"%1.4f %1.4f\n",cosTETA,EgU);
   cosTETA += 0.01;
}while (cosTETA < 1.0);
/**/
}
```