Temat:			
Metoda Monte Carlo			
Wykonał: Marcin Fabrykowski	Wydział: FiIS	Kierunek Inf. Stos.	Grupa: grupa 3

1. Wstęp
   Metoda Monte Carlo pozwala uprościć obliczenia matematyczne gdy są one skomplikowane, bądź niemożliwe do obliczenia w skończonym czasie, kosztem dokładności obliczeń. Polega ona na losowaniu próbek i sprawdzaniu tych próbek ze znanymi zależnościami. Przykładem mogą być symulacje zderzeń cząstek, gdzie losowane są parametry zderzenia, prędkość, kąt a następnie ze znanych zależności, obliczany jest wynik takiego zderzenia. Zakładając losowość generatora pseudolosowego można powiedzieć ze otrzymane wyniki są prawdopodobne. Innym przykładem może być obliczanie momentu bezwładności bryły. Tutaj również posługujemy się losowymi próbkami punktów w danym obszarze.
   Wyniki otrzymywane tą metodą są szybkie ale cechują się pewnym błędem wynikającym z losowości próbek

2. Wykonanie ćwiczenia
   Celem ćwiczenia jest wyznaczenie momentu bezwładności wydrążonego cylindra o promieniach $r_1 = 0.5$ oraz $r_2 = 0.7$ wzdłuż jego osi obrotu.
   Moment bezwładności obliczamy ze wzoru: $I = \int\limits_M r^2 dm$. Zakładamy jednorodność bryły: $dm = \sigma dV$ co daje:

$$I = \sigma \int\limits_{\Omega} r^2 d\Omega$$

   Do metody Monte Carlo, będziemy potrzebowali wzoru który będzie uwzględniał trafienie próbki w badany obszar.

$$I = \frac{V\sigma}{N} \sum_{n=1}^{N} r_i^2 \Theta_i$$

   gdzie N to liczba próbek, V objętość obszaru w którym będziemy losować próbki, Θ jest boolowską funkcją sprawdzającą czy dana próbka znajduję się w obszarze Ω będącym obszarem badanej przez nas bryły.
   Dodatkowa wiedza która będzie nam potrzebna, to jak policzyć odległość punktu od prostej.
   Zauważamy ze odległość od osi obrotu, możemy policzyć tylko na podstawie składowych x i y:

$$r = \sqrt{x^2 + y^2}$$

   W naszym ćwiczeniu założymy gęstość $\sigma = 1$ oraz liczbę próbek $N \leqslant 10^6$. Obszar V przyjmujemy jako sześcian o boku $a = 2$ natomiast Ω jest

wydrążonym cylinder o parametrach podanych na początku podpunktu.
Następnie obliczamy błąd oszacowanie

$$s^2(N) = \frac{1}{N-1}\left[\sum_{i=1}^{N}(V\cdot\sigma\cdot r_i^2\cdot\Theta_i)^2 - \frac{1}{N}\left(\sum_{i=1}^{2}V\cdot\sigma\cdot r_i^2\cdot\Theta_i\right)^2\right]$$

Natomiast odchylenie standardowe:

$$s(I) = \sqrt{\frac{s^2}{N}}$$

Program rozwiązujący powyższy problem ma kod:

Listing 1: main.cpp

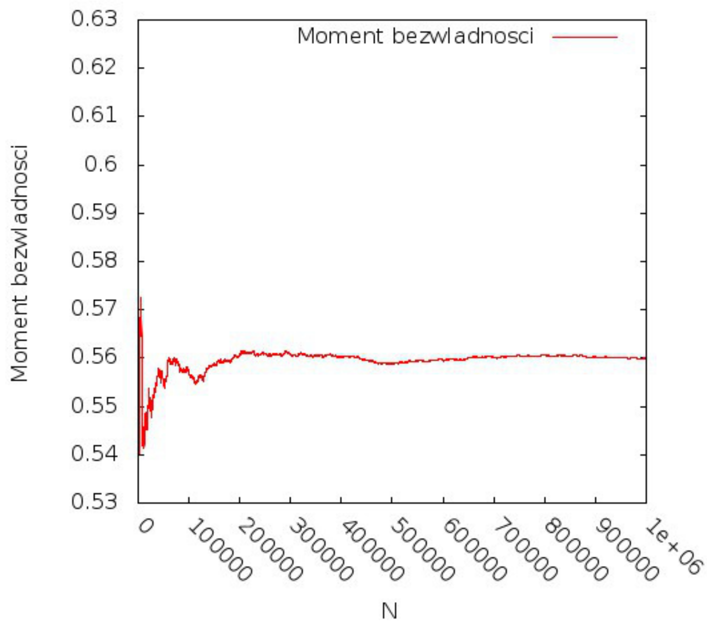
```
#include <iostream>
#include <stdlib.h>
#include <cmath>
using namespace std;
bool theta(float x,float y, float z);
const float r1=0.5;
const float r2=0.7;
const float h=1;
float d(float x,float y);
float los(float a);
int main()
{
    int N_MIN=1000;
    int N_MAX=1000000;
    srand(time(NULL));

    float a=2; //Bok obszaru V;
    float V=a*a*a;
    int N=N_MAX;
//  for(N=N_MIN;N<N_MAX;N+=100)
    {
            int i;
            float I=0;
            float sum1=0;
            float sum2=0;
            for(i=0;i<N;i++)
            {
                float x=los(a)-a/2;
                float y=los(a)-a/2;
                float z=los(a/2)-a/4;
        //      cout<<"Sprawdzam punkt
            "<<x<<","<<y<<","<<z<<", i stwierdzam ze thera
            wynosi: "<<theta(x,y,z)<<endl;
                if(theta(x,y,z))
                {
        //          cout<<"jest"<<endl;
                    I+=d(x,y)*d(x,y);
                    float tmp=V*d(x,y)*d(x,y);
                    sum1+=tmp*tmp;
                    sum2+=tmp;
                };
                if(!(i%100))
                {
                    cout<<i<<" "<<I*V/i<<" 
                        "<<sqrt(((sum1-(sum2*sum2)/i)/(i-1)/i))<<endl;
```

```
                    };
                };
                I*=V;
                I/=N;
            //  cout<<"Moment bezwladnosci wynosi: "<<I<<endl;
                float s2=sum1-(sum2*sum2)/N;
                s2/=(N-1);
                float s=sqrt(s2/N);
            //  cout<<"Blad s^2="<<s2<<"\ts="<<s<<endl;
//              cout<<N<<" "<<I<<" "<<s<<endl;
    };
/*

    V szescian o rymiarach 1x1x1
    I=V*s/N sun r^2*tetha;

    s^2=1/(N-1) [ sum\limits_{i=1}^N
        (V*\sigma*r_i^2*\Theta_i)^2-1/N(\sum\limits_{i=1}^N
        V*\sigma*r_i_^2*\Theta)^2
    d=\sqrt((|r1-ri|)^2|r2-r1|^2-[(r1-ri)(r2-r1)]^2 /
        (|r2-r1|^2))

    I=1/2 m (R1^2+R2^2);
    V=V*sigma;
    V=pi(R2^2-R1^2)*1=0.7536
    m=0.7536
    I=1/2*0.7536*(0.5^2+0.7^2)=0.5842
*/
};
bool theta(float x,float y, float z)
{
    float r=d(x,y);
//  cout<<"r="<<r<<endl;
    if(z>h/2||z<-h/2) return false;
    if(r<r1||r>r2) return false;
    return true;
};
float los(float a=1)
{
    return a*(float)rand()/(RAND_MAX);
};
float d(float x,float y)
{
    return sqrt(x*x+y*y);
};
```

Wynik jego działania można przestawić graficznie:

3. Wnioski
   Powyższe wyniki otrzymywane są w krótkim czasie a dokładność ich jest zadowalająca w stopniu inżynierskim. Oczekiwany moment bezwładności wyliczony ze wzoru dokładnego wynosi: $0,55794686$ natomiast otrzymamy metodą Monte Carlo wynosi: 0.560013, co w rozsądnym inżynierskim przybliżeniu jest wynikiem zadowalającym.
   Powyższy przykład pokazuję zalety tej metody, łączące szybkość otrzymywanych wyników z ich dokładnością.

Rysunek 1: Moment bezwładności



Rysunek 2: Błąd oszacowania