

Programowanie

Ćwiczenia, semestr letni 2023

Michał Suchorowski

20 marca 2023

Spis treści

1 Ćwiczenia 3	1
1.1 Ćwiczenie 1	1
1.2 Ćwiczenie 2	1
1.3 Ćwiczenie 3	1
1.4 Ćwiczenie 4	1
1.5 Dodatkowe ćwiczenie 1	2

1 Ćwiczenia 3

1.1 Ćwiczenie 1

Symulacja rzutu kostką. Potrzebna biblioteka `cstdlib` i funkcje `rand`, `srand` i `time`.

1.2 Ćwiczenie 2

Symulacja rzutu kostką złożoną z wszystkich liter alfabetu. Zobacz <https://pl.wikipedia.org/wiki/ASCII>.

1.3 Ćwiczenie 3

Oszacuj wartość liczby π poprzez policzenie stosunku pola kwadratu do koła.

```
#include <cstdlib>
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int N=100000;
    srand( time( NULL ) );
    double x,y,r;
    int counter=0;
    for(int i=0; i<N; i++){
        x=(double)rand()/RAND_MAX*2-1;
        y=(double)rand()/RAND_MAX*2-1;
        r=sqrt(x*x+y*y);
        if(r<1)
            counter++;
    }
    cout << (double)counter*4/N << endl;

    return 0;
}
```

1.4 Ćwiczenie 4

Napisz program, który całkuje funkcję $f(x) = y$ metodą Monte Carlo w zakresie od a do b . Wylosuj x oraz y_r (y_r losuj z zakresu między y_{min} - y_{max} w którym mieści się wykres funkcji). Następnie sprawdź czy $y_r < y$. Stosunek ilości punktów pod wykresem do wszystkich losowań będzie równy stosunkowi pola pod wykresem oraz prostokątowi $(b - a) * (y_{max} - y_{min})$.

```

#include <cstdlib>
#include <iostream>
#include <cmath>

#include <math.h>

using namespace std;

int main()
{
    int N=1000000;
    srand( time( NULL ) );

    double a=-M_PI*2;
    double b=M_PI*2;
    double ymin=0;
    double ymax=1;

    double x,y;

    int counter=0;
    for(int i=0; i<N; i++){
        x=(double)rand()/RAND_MAX*(b-a)+a;
        y=(double)rand()/RAND_MAX*(ymax-ymin)+ymin;
        if ( y < sin(x)*sin(x) )
            counter++;
    }

    double S = (double)counter/N * ( (b-a)*(ymax-ymin) );
    cout << S/M_PI << "_Pi" << endl;
    return 0;
}

```

1.5 Dodatkowe ćwiczenie 1

Rozwiń program z ćwiczenia 3 tak, żeby policzyć stosunek objętości N -wymiarowej kuli do ograniczającej jej prostopadłościanu. Wypisz po kolei ten stosunek od $N = 1$ do $N = 10$ i zobacz jak ten stosunek zmienia się wraz z N . (Dla $N = 1$ ten stosunek powinien wynosić 1).

```

#include <cstdlib>
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int N=100000;
    srand( time( NULL ) );
    double r,x;

    for(int n=1; n<15; n++){
        int counter=0;
        for(int i=0; i<N; i++){
            r=0;
            for(int j=0; j<n; j++){
                x=(double)rand()/RAND_MAX*2-1;
                r+=x*x;
            }
            if(r<1)
                counter++;
        }
        cout << (double)counter/N << endl;
    }

    return 0;
}

```