# LAPORAN TUGAS

# PROGRAM INTEGRAL MONTE CARLO
# DASAR PEMODELAN DAN SIMULASI

Disusun Oleh :


| | |
|---|---|
| Taufik Fathurahman | 1301160790 |
| Haddad Alwi Yafie | 1301162756 |
| Alanu Dinasti Permana | 1301160774 |

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**UNIVERSITAS TELKOM**

**BANDUNG**


**2019**

# I. Paparan Masalah

Dalam laporan ini, terdapat dua buah fungsi yang akan diuji, yaitu:

$$f(x) = \int_{-5}^{5} \frac{1}{\sqrt{2\pi}} exp^{-\frac{x^2}{2}}$$

$$f(x) = \int_{-5}^{5} 3x^2$$

Kedua fungsi tersebut akan dihitung nilai integralnya secara numerik dan eksak. Untuk perhitungan secara numerik akan dipecahkan dengan teknik pengintegralan metode *Monte Carlo* alternative satu dan dua. Parameter yang diujikan terhadap kedua fungsi diatas adalah sama, yaitu akan dilakukan pengujian dengan nilai N sebesar 10000, dengan selang (1000:1000:N) sehingga aka nada sepuluh solusi nilai integral numerik. Untuk alternative satu akan ada parameter k sebesar 1,5, dan untuk alternative dua aka nada parameter fmin senilai 0 dan fmax.

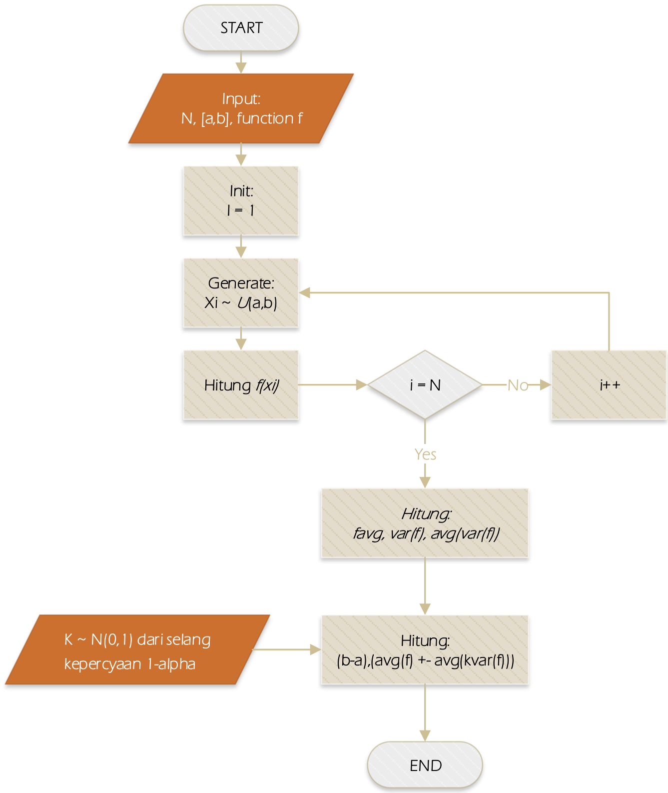
## II. Metode Monte Carlo untuk menghitung Integral numerik

Monte Carlo merupakan suatu metode yang menggunakan bilangan acak untuk mengambil titik sampel dari daerah yang diintegrasikan. Salah satu metode yang unik dalam metode ini adalah galat yang diperoleh dari metode integrase ini bisa berbeda tergantung dari jumlah datanya. Simulasi Monte Carlo sering digunakan untuk melakukan analisa keputusan pada situasi yang melibatkan resiko yang melibatkan beberapa parameter untuk dilakukan pertimbangan secara simultan. Metode ini dapat digunakan secara luas karena didasarkan pada proses simulasi dengan pilihan kemungkinan secara *random*. Dengan demikian, jumlah iterasi yang dilakukan sangat menentukan tingkat ketelitian atas jawaban yang diperoleh. Metode ini seringkali juga disebut dengan metode percobaan statistik.

Untuk menghitung nilai integral dengan menggunakan metode Monte Carlo dibutuhkan suatu pembangkit bilangan acak dimana terdapat masalah juga dalam bagaimana bilangan semu-acak yang dihasilkan oleh komputer dapat memenuhi kebutuhan tersebut. Salah satu penerapan yang paling penting dari Monte Carlo adalah pada fenomena partikel dimana Monte Carlo menghasilkan titik-titik pada ruang fase multipartikel.

Dalam menyeselesaikan menyeselaikan kedua fungsi di atas, berikut ini desain algoritma *monte carlo* yang dibuat:

a. Monte Carlo 1

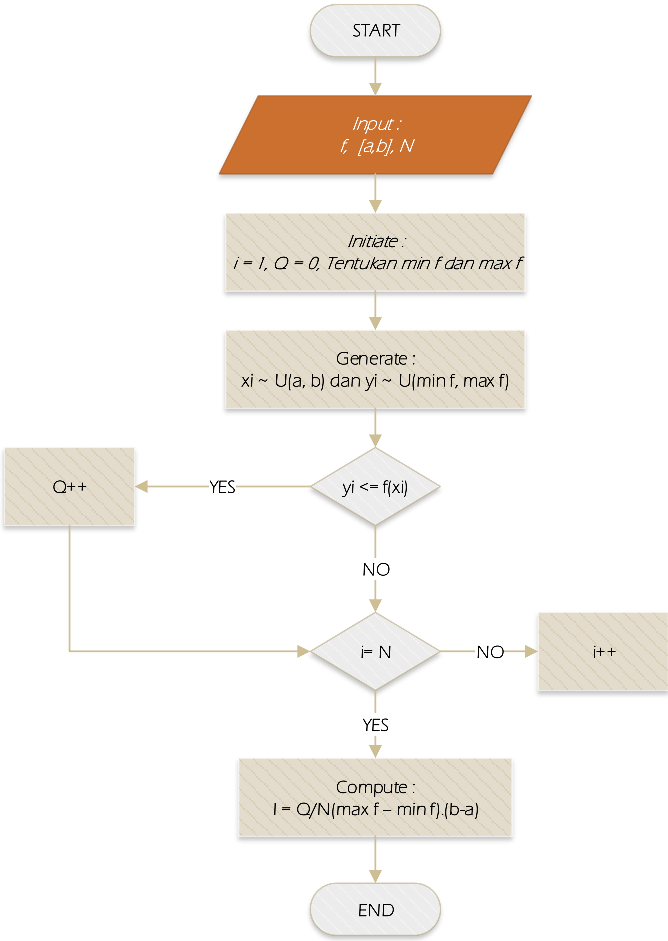



*Flowchart 1Monte Carlo 1*

```
function montecarlo1(handles, function_fx, batas_atas, batas_bawah, N, K)
fx = function_fx;
interval_integral = [];
hasil_mean = [];
for i = 1000:1000:N
    hasil_fx = [];
    rand_num = batas_bawah + (batas_atas - batas_bawah).*rand(i,1);
    for j = 1:i
        x = num2str(rand_num(j));
        x = ['(' x ')'];
        new_fx = strrep(function_fx, '(x)', x);
        hasil_fx =[hasil_fx eval(new_fx)];
    end
    hasil_avg = (1/i)*sum(hasil_fx);
    varian = (1/(i-1))*sum(hasil_fx)-hasil_avg;

    nilai_integral1 = (hasil_avg+K*varian)*(batas_atas-batas_bawah);
    nilai_integral2 = (hasil_avg-K*varian)*(batas_atas-batas_bawah);

    p = ['(' num2str(nilai_integral2) ' <=> '];
    q = [num2str(nilai_integral1) '), '];

    interval_integral = [interval_integral [p q]];

    mean_integral = (nilai_integral1 + nilai_integral2)/2;
    hasil_mean = [hasil_mean mean_integral];
end
end
```

*Gambar 1 Algoritma Monte Carlo 1*

## b. Monte Carlo 2




*Flowchart 2Monte Carlo 2*

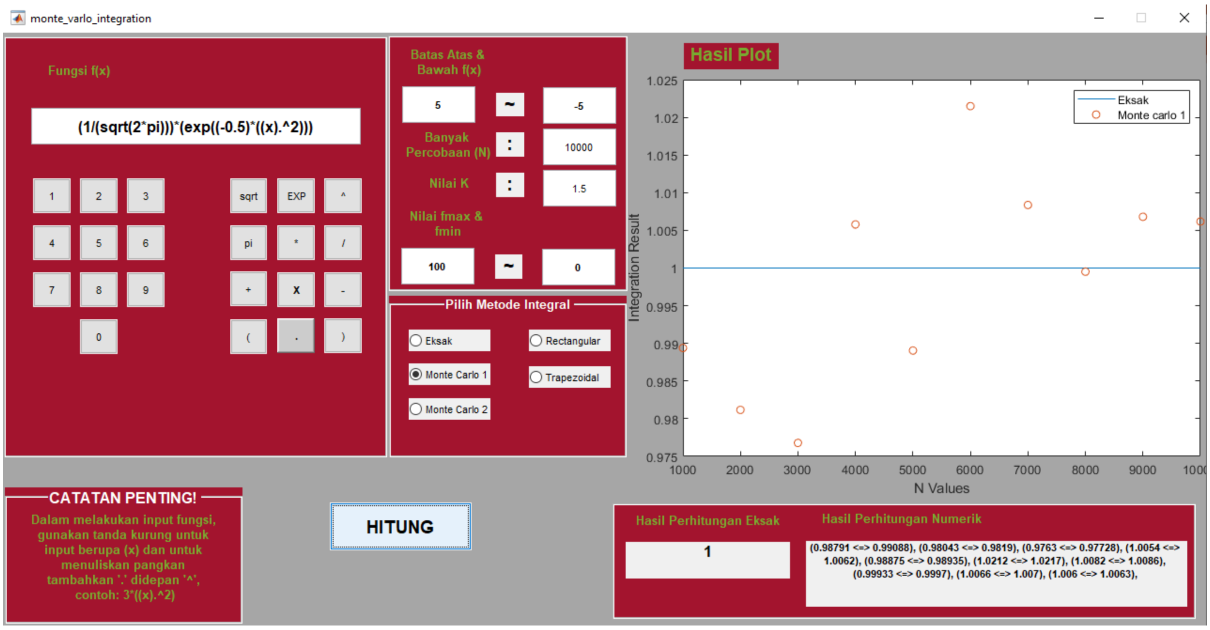
```
function montecarlo2(handles, function_fx, batas_atas, batas_bawah, f_min, f_max, N)

fx = function_fx;
hasil_integral = [];

for i = 1000:1000:N
    Q = 0;
    for j = 1:i
        Xi = batas_bawah + (batas_atas - batas_bawah).*rand();
        Yi = f_min + (f_max - f_min).*rand();

        x = ['(' num2str(Xi) ')'];
        new_fx = strrep(function_fx, '(x)', x);
        hasil_fx = eval(new_fx);

        if hasil_fx >= Yi
            Q = Q + 1;
        end
    end
    I = (Q/i)*(f_max-f_min)*(batas_atas-batas_bawah);
    hasil_integral = [hasil_integral I];
end

end
```

*Gambar 2 Monte Carlo 2*

# III. Hasil dan Penjelasan

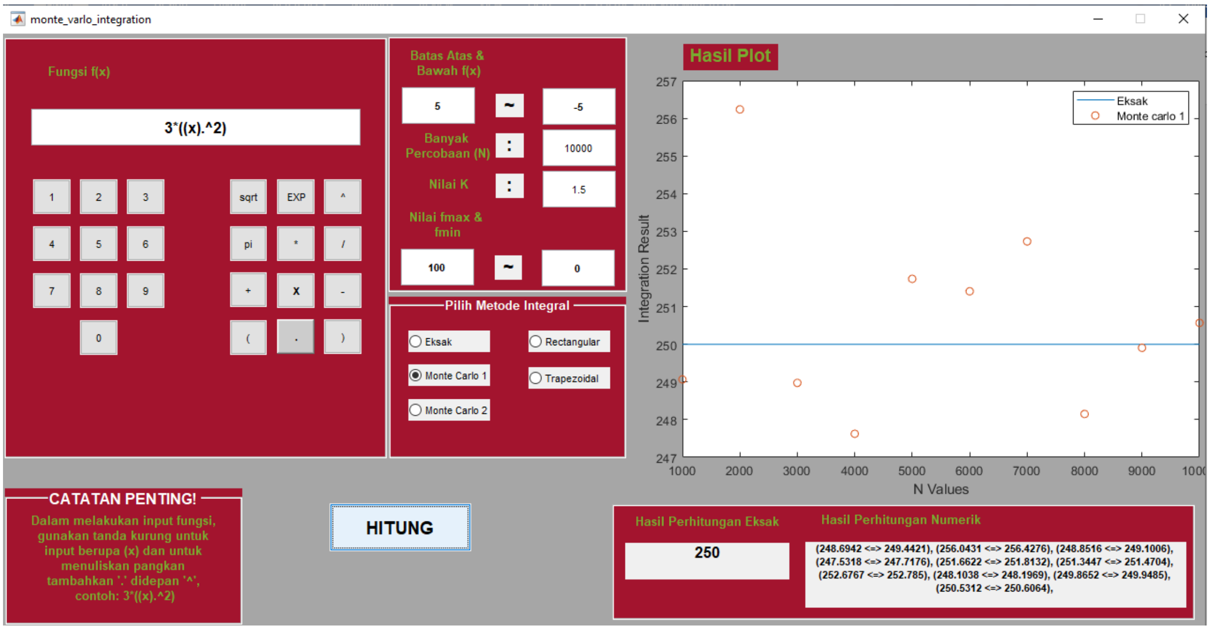
## a. Monte Carlo 1

### 1. Function 1




*Gambar 3 Hasil MC1 function 1*

### 2. Function 2



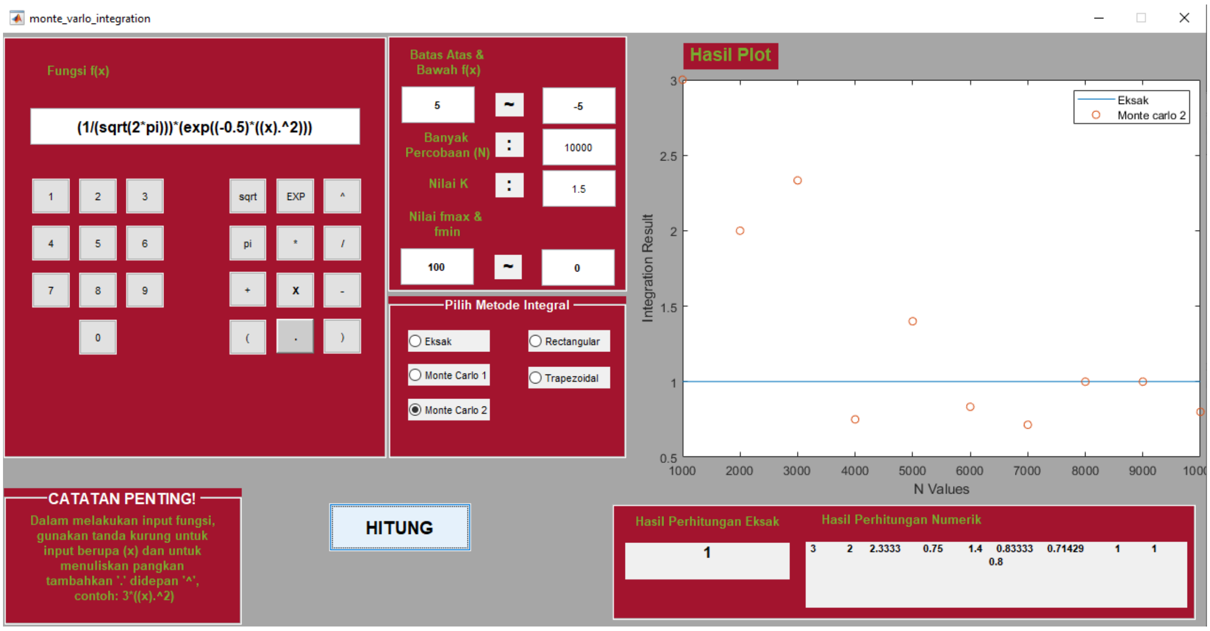

*Gambar 4 Hasil MC1 function 2*

Analisis :

Dari percobaan monte carlo 1 ini menghasilkan dua jenis output yaitu plot dan array. Untuk plot, kami melakukan penyederhanaan dengan mengambil nilai rata-rata dari nilai selang hasil perhitungan untuk mempermudah melakukan plot. *Output* array berupa nilai selang dari solusi integralnya. Dari percobaan sepuluh kali dengan nilai N 1000 sampai 10000 dengan kenaikan 1000 didapatkan hasil integral numerik yang cenderung semakin mendekati nilai eksaknya. Selain nilai N, yang memberikan pengaruh dalam mendapatkan solusi numerik adalah nilai f minimum dan f maksimumnya.

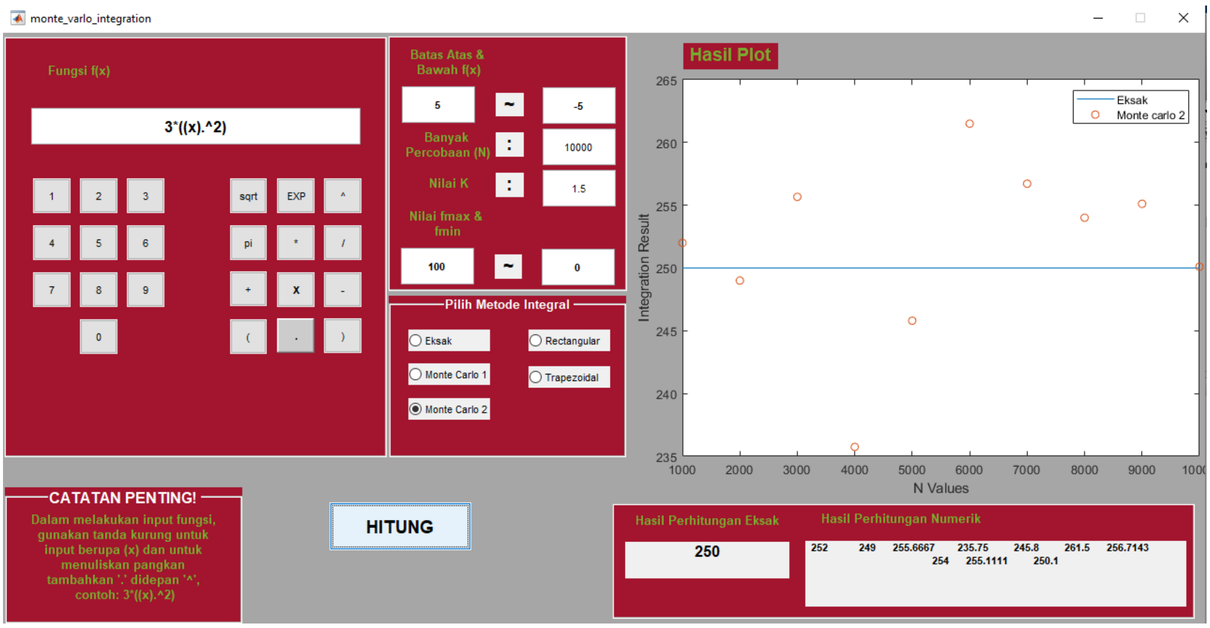
b. Monte Carlo 2

1. Function 1




*Gambar 5 Hasil MC2 function 1*

2. Function 2




*Gambar 6 Hasil MC2 function 2*

Analisis:

Pada percobaan metode monte carlo yang kedua ini juga menghasilkan dua jenis *output*, yaitu plot dan *vector* dari sepuluh solusi integral. Dari percobaan sepuluh kali dengan nilai N 1000 sampai 10000 dengan kenaikan 1000 didapatkan hasil integral numerik yang cenderung semakin mendekati nilai eksaknya juga.

## IV. Kesimpulan

Dalam simulasi *monte carlo* setiap titik dari sampel dan daerah dibawah fungsi f(x) menggunakan bilangan acak dengan selang yang telah ditentukan. Banyak permasalahan yang dapat diselsaikan dengan menggunakan metode *monte carlo,* yang salah satunya adalah menghampiri solusi untuk nilai integral.

Berdasarkan percobaan yang telah dilakukan dengan menggunakan metode *monte carlo* terhadap dua fungsi, didapati bahwa solusi numerik akan cenderung semakin mendekati solusi eksak bilai nilai N-nya semakin besar. Namun bukan berarti bila kita menggunakan nilai N yang kecil akan mendapatkan nilai numerik yang buruk, karena tidak jarang juga dengan nilai N yang kecil dapat mendekati nilai eksak dengan besar error atau galat yang kecil. Selain factor dari penggunaan nilai N, variable lain juga sangat menentukan dalam mendapatkan nilai numerik integral yang presisi, seperti nilai k, f min, dan f maks.