**LAPORAN PRAKTIKUM**

**METODE NUMERIK**

**Judul: Galat/Error**



**DISUSUN OLEH**

**ILHAM NUR ROMDONI M0520038**

**PROGRAM INFORMATIKA**

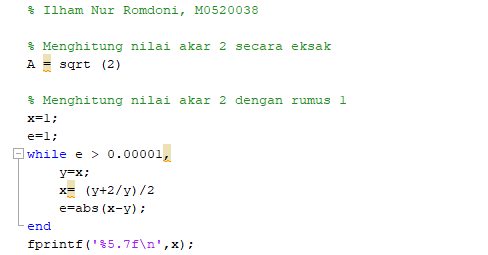
**FAKULTAS MIPA**

**UNIVERSITAS SEBELAS MARET**

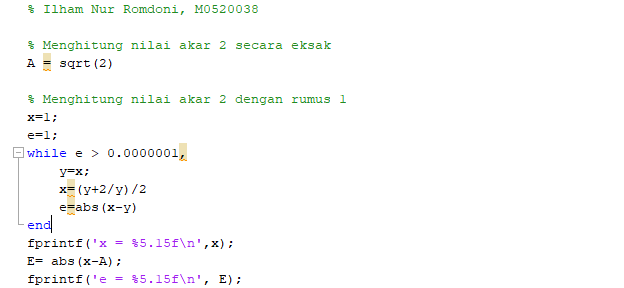
**2021**

**SCREENSHOT**

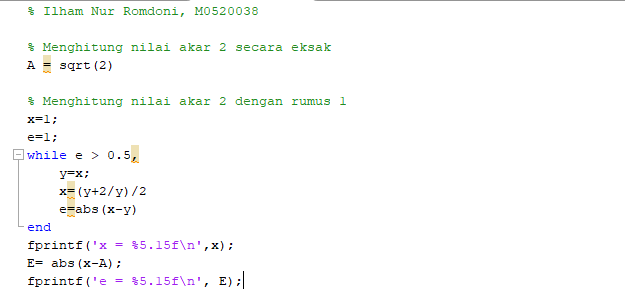
1. **Screenshot Praktikum**
2. **Program 1: Menghitung nilai**

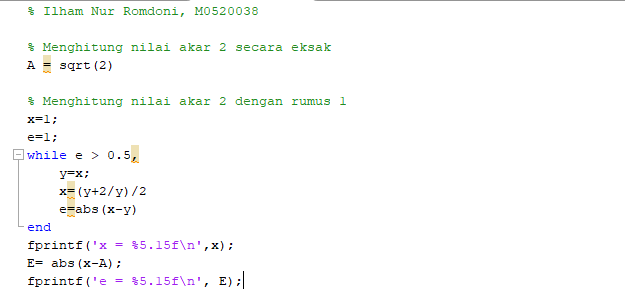


1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode.

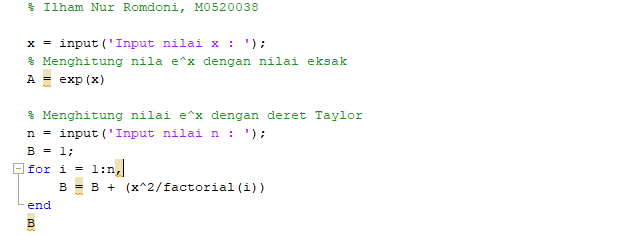


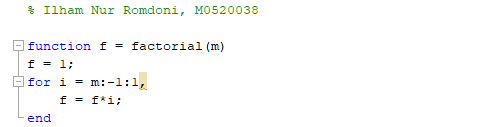
1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode jika batas iterasi menjadi 0,5.



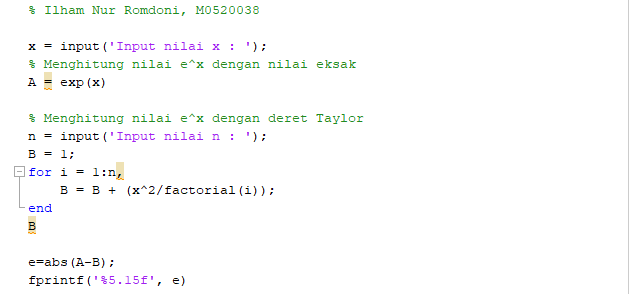


1. **Program 2: Menghitung**

****

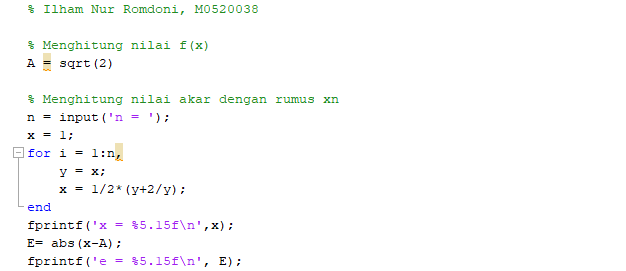


1. Menghitung nilai *error* dari nilai dengan Deret Taylor, misalkan n = 10 atau 5.



1. **Screenshot Source Code**
2. **Soal No 1**

Tentukan nilai galat atau *error* pada perhitungan nilai f(x) = dengan rumus dengan dan dengan .

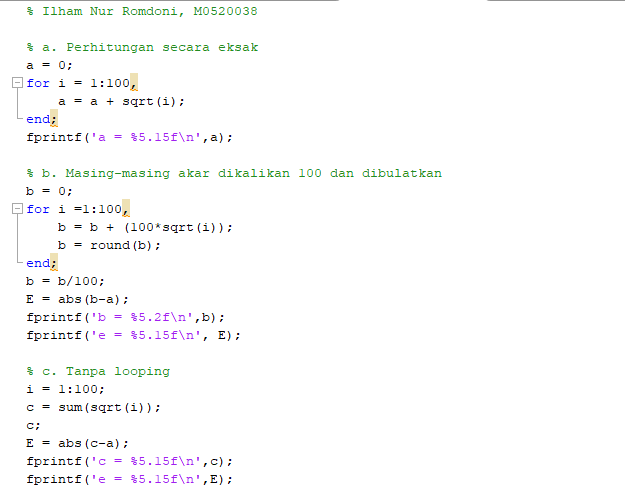
****

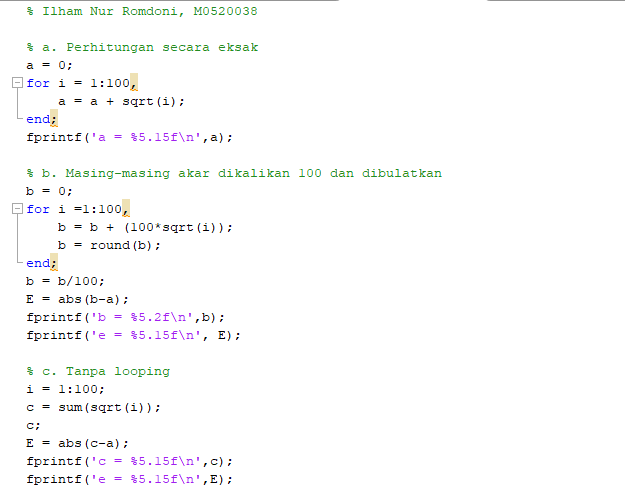
1. **Soal No 2**

Tentukan nilai galat atau *error* pada perhitungan nilai

dengan ketiga metode di bawah ini:

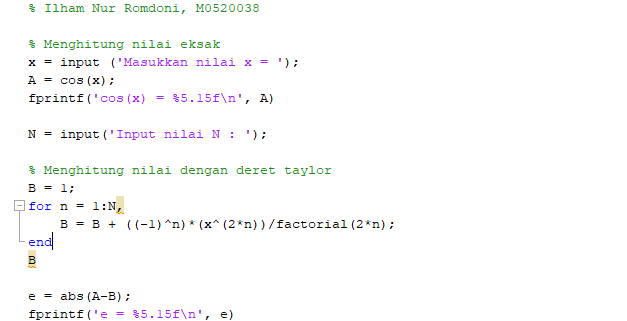
1. Perhitungan secara eksak.
2. Masing-masing akar dikalikan 100 dan dibulatkan.
3. Tanpa *looping* ( Menggunakan fungsi SUM dalam MATLAB ).





1. **Soal No 3**

Tentukan nilai galat atau *error* pada perhitungan nilai dengan menggunakan deret Taylor yang dapat dirumuskan di bawah ini:



**ANALISIS PRAKTIKUM**

1. **Analisis Source Code**
2. **Program 1: Menghitung nilai**

Untuk menghitung nilai eksak dari pada MATLAB, menggunakan fungsi sqrt. Jadi nilai variabel A akan diisi oleh hasil dari sqrt(2). Selanjutnya untuk melakukan penghitungan nilai dengan rumus 1, dimulai dengan mendefinisikan nilai awal yaitu x=1 dan e =1. Setelah itu melakukan pengulangan while di mana syarat e > 0.00001. Ketika e masih > 0.00001 maka pengulangan while masih berjalan. Variabel y diinisialisasikan dengan nilai x sedangkan nilai x diisi dengan nilai baru dari rumus (y+2/y)/2. Nilai e dihitung dengan abs(x-y). abs adalah fungsi untuk nilai mutlak. Fungsi fprintf digunakan untuk menampilkan nilai dari variabel. %5.7f berarti angka 5 adalah jumlah angka di depan koma desimal. Sedangkan 7 adalah jumlah angka di belakang koma dan huruf f merepresentasikan bahwa bilangan ini bertipe data float.

1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode.

Mendefinisikan variabel E sebagai nilai galat dari eksak dengan rumus 1. E = abs(x-A) yaitu nilai mutlak dari pengurangan nilai eksak dan nilai rumus pada iterasi terakhir.

1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode jika batas iterasi menjadi 0,5.

Nilai e pada pengulangan while diganti dengan 0,5.

1. **Program 2: Menghitung**

Nilai dihitung menggunakan fungsi exp(x) di mana nilai x diambil dari *input*-an *user*. *Input*-an *user* dapat disimpan dengan menggunakan fungsi input(). Penghitungan pendekatan dilakukan dengan deret Taylor. n didefinisikan sebagai batas pengulangan yang akan dilakukan for. Variabel B memiliki nilai awal yaitu 1. Pengulangan menggunakan for di mana didefinisikan pengulangan i = 1 sampai n kali. Variabel B akan diisi dari hasil rumus B + (x^2/factorial(i)) yang mana nilai terus bertambah dan nilai B akan berganti terus sesuai nilai i. Fungsi factorial bukan bawaan dari MATLAB. Fungsi factorial dibuat sendiri dengan mendefinisikan f sebagai factorial(m). Setelah menentukan nilai awal, melakukan pengulangan for dengan i = m, setiap perulangan berkurang 1. Yaitu dengan menuliskan for I = m:-1:1. Lalu mendefinisikan variabel f dengan f\*i.

1. Menghitung nilai *error* dari nilai dengan Deret Taylor, misalkan n = 10 atau 5.

Mendefinisikan variabel e sebagai nilai galat dari eksak dengan deret Taylor. e = abs(A-B) yaitu nilai mutlak dari pengurangan nilai eksak dan nilai deret Taylor pada iterasi terakhir. Nilai x yang di-*input*-kan adalah 1 dan nilai n yang di-*input*-kan adalah 10 atau 5.

1. **Soal No 1**

Pendefinisian nilai awalnya, yaitu x = 1. Memberikan *input* ke nilai n. Pendekatan dihitung menggunakan perulangan i yang dimulai dari 1 hingga berhenti di n (sesuai angka *input*-an). Menginisialisasikan variabel y = x. Nilai x diisi dengan nilai yang baru dari rumus ½\*(y+2/y). Nilai e yang mendefinisikan *error*, diisi dengan nilai absolut (mutlak) dari x dikurangi y. Nantinya akan menampilkan hasil dari *error* pada perulangan terakhir.

1. **Soal No 2**

Pendefinisian a = 0. Perulangan i di mana akan mengalami perulangan hingga 100 kali. Inisialisasi a bahwa nantinya variabel a ditambah dengan akar dari perulangan saat itu (i). Menampilkan hasil dari a dengan bilangan desimal 15 digit di belakang koma.

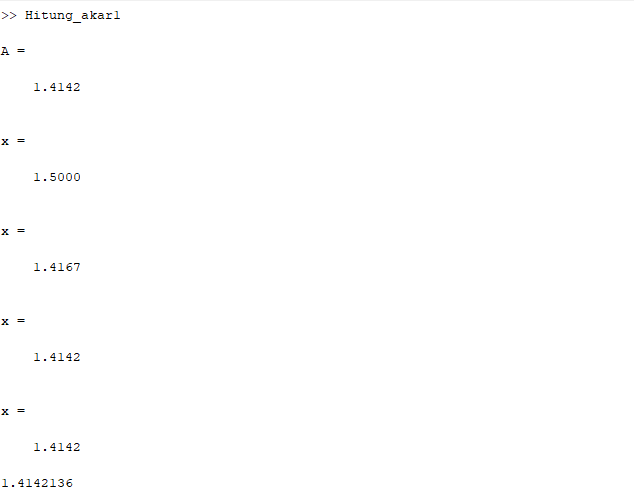
Pendefinisian b = 0. Perulangan i di mana akan mengalami perulangan hingga 100 kali. Inisialisasi b bahwa nantinya variabel b ditambah dengan perkalian antara bilangan 100 dan akar i. Kemudian variabel b dibulatkan menggunakan fungsi round. Perhitungan *error* dengan absolut dari variabel b dikurangi variabel a. Menampilkan hasil b yang berupa desimal dengan 2 digit di belakang koma dan hasil *error* dari b.

Pendefinisian bahwa i nantinya adalah bilangan dari 1 hingga 100. Inisialisasi c sama dengan penjumlahan (sum) dari akar i di mana i tersebut adalah bilangan dari 1 hingga 100. Perhitungan *error* dengan absolut dari variabel c dikurangi variabel a. Menampilkan hasil c dan hasil *error* dari c.

1. **Soal No 3**

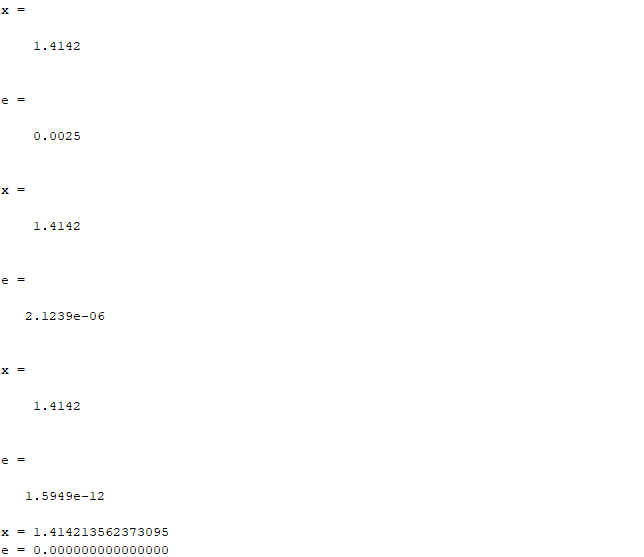
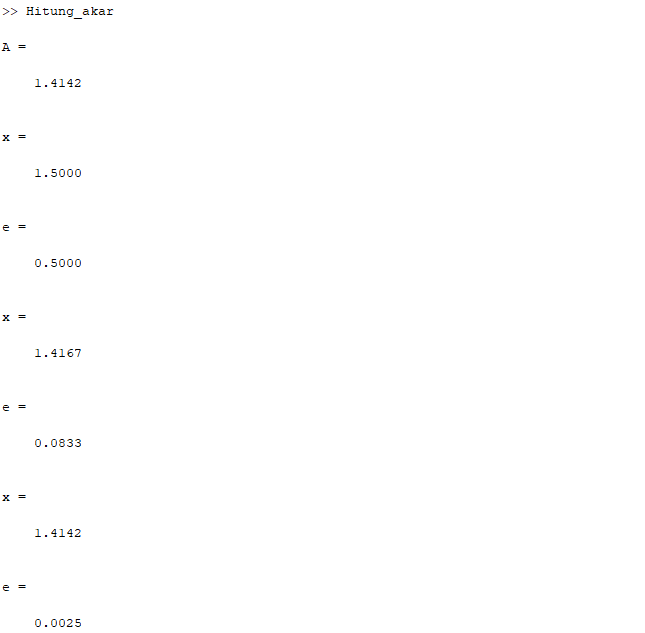
Memberikan *input* pada x, dengan fungsi *input*. Pendefinisian variabel A = cos(x) di mana x-nya di dapat dari *input*-an *user*. Kemudian hasil tersebut ditampilkan ke *Command* *Window* menggunakan fungsi fprintf. Kemudian memberikan *input* pada N. pendefinisian bahwa B = 1 di mana angka awal untuk memulai deret taylor tersebut. Melakukan perulangan n di mana dari 1 hingga ke N (diambil dari *input*-an N dari *user*). Selama perulangan 1 ke N akan menjalankan fungsi B = B + ((-1)^n)\*(x^(2\*n)). Kemudian nilai B ditampilkan. Menghitung nilai *error* dengan absolut dari variabel A dikurangi dengan variabel B.

1. **Analisis Jalannya Program**
2. **Program 1: Menghitung nilai**

****

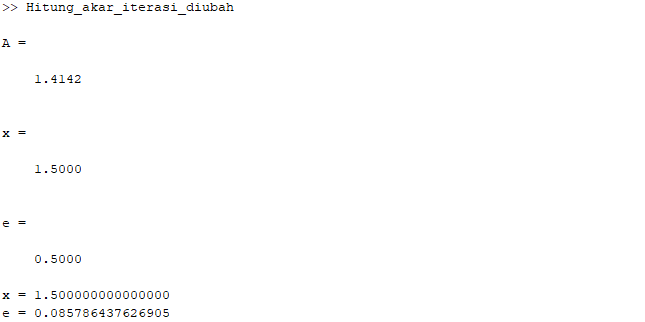
Pada program asli praktikum, menampilkan hasil dari akar 2 dan nilai x dengan batasan error yaitu 0.00001.

1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode.

****

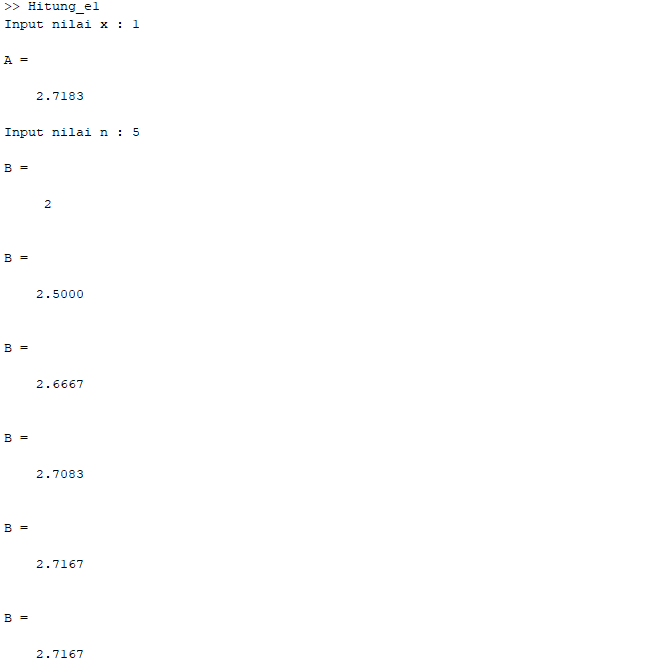
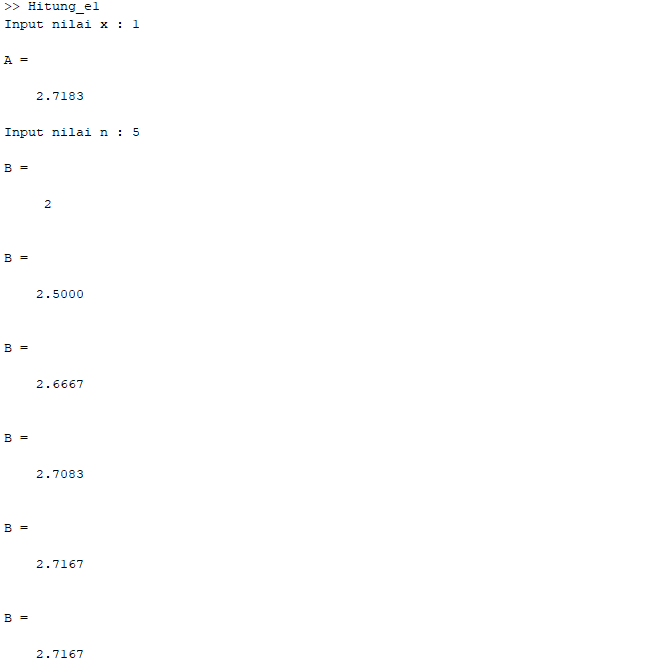
Program yang dibagikan pada *google* *classroom*, menggunakan batasan *error* 0.0000001 dan menampilkan hasil x dengan 15 angka desimal. Nilai e = 0.

1. Menghitung nilai galat dari nilai eksak dengan nilai dari metode jika batas iterasi menjadi 0,5.



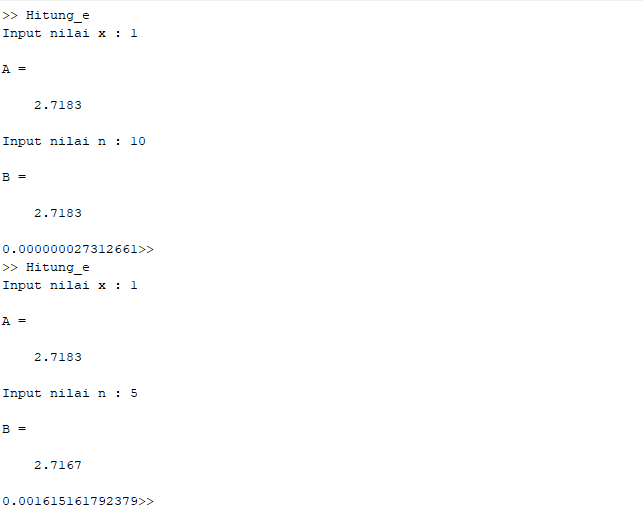
*Error* semakin besar karena batas iterasi 0.5 sehingga proses iterasi berlangsung sedikit.

1. **Program 2: Menghitung**

****

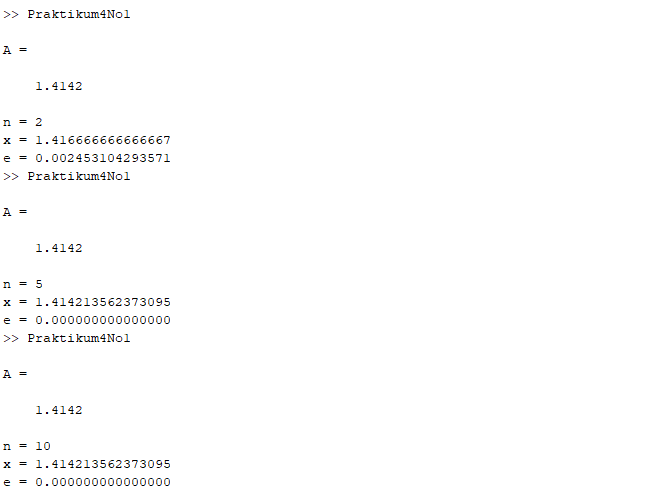
Pada program asli praktikum, menampilkan hasil dari nilai exp(x) dan nilai dari deret Taylor sampai iterasi ke-n.

1. Menghitung nilai *error* dari nilai dengan Deret Taylor, misalkan n = 10 atau 5.



Dengan n=10 memberikan error 0.000000027312661 sedangkan dengan n=5 memberikan nilai error 0.001615161792379. Semakin banyak iterasi, semakin kecil *error*-nya.

1. **Soal No 1**

****

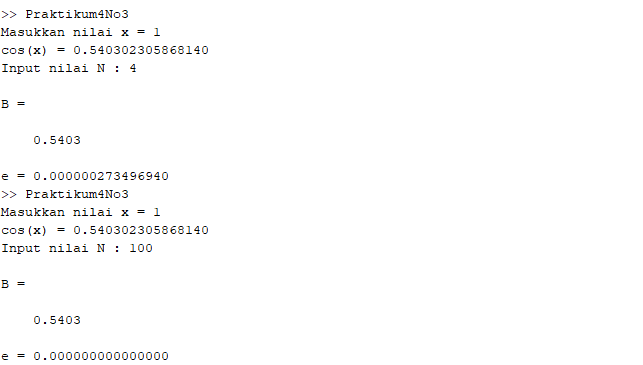
Program menampilkan nilai *error* dari soal no 1. Ketika n=2, *error* bernilai 0.002453104293571. Sedangkan saat n=5 dan n=10, *error* bernilai 0. Hal ini membuktikan bahwa semakin banyak perulangan, maka nilai *error*-nya semakin kecil bahkan mendekati tanpa *error*. Nilai e = 0 menunjukkan batas maksimal iterasi.

1. **Soal No 2**

****

Hasil dari proses perhitungan ditunjukkan bahwa hasil *error* pada 2b yaitu 0.017052896852306 dengan rumus mutlak dari nilai eksak pada poin a dikurangi dengan nilai pendekatan dengan *rounded* atau pembulatan. Sedangkan pada 2c *error* sama dengan nol karena hasil dari perhitungan tanpa perulangan sama dengan nilai eksak sehingga jika dikurangi akan menghasilkan nol.

1. **Soal No 3**

****

Hasil dari percobaan tersebut bahwa pada iterasi 4 memiliki *error* 0.000000273496940 dan iterasi 100 memiliki *error* yang sangat mendekati 0. Sehingga semakin banyak iterasi maka *error* dari suatu perhitungan akan semakin kecil. Nilai e = 0 menunjukkan batas nilai maksimal dari N atau iterasi penyelesaian.