**PRAKTIKUM MANDIRI METODE NUMERIK 2019**

**TURUNAN DAN INTEGRAL NUMERIK**

**NAMA : WAHYU ALFANDI**

**NIM : F1D018060**

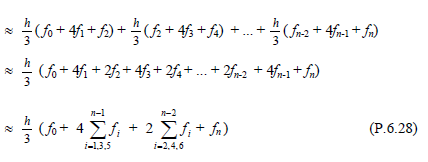
1. **TUJUAN PRAKTIKUM**
2. Mahasiswa mampu memahami turunan numerik dengan selisih maju, mundur, dan terpusat kemudian dapat mengimplementasikannya ke dalam program C++.
3. Mahasiswa mampu memahami dan mengimplementasikan metode newton code 1/3 dalam bahasa C++.

Diberikan data x dan y sbb:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x | -1 | -0.5 | 0 | 0.5 | 1 | 1.5 | 2 | 2.5 | 3 | 3.5 | 4 |
| y | 0 | 3.375 | 4 | 2.625 | 0 | -3.125 | -6 | -7.875 | -8 | -5.625 | 0 |

Berdasarkan data di atas, tentukan :

1. Turunan pertama dengan selisih maju, mundur, dan terpusat pada x=0.75, dengan h=0.01, bandingkan ketiganya.
2. Hitunglah integral dari data yang diberikan dengan NC 1/3 dari -1 s.d 2 dengan h=0.01
3. **ALGORITMA PENYELESAIAN**
4. **Rumus turunan pertama**
5. Selisih maju
6. Selisih mundur
7. Selisih pusat
8. **Rumus Integral dengan metode NC – Simpson 1/3**

****

*Source code :*

|  |
| --- |
| #include<iostream>  #include<cmath>  using namespace std;  double interpolasi (double a){      int j,i,n=11;      double x[] = {-1,-0.5,0,0.5,1,1.5,2,2.5,3,3.5,4};      double f[] = {0,3.375,4,2.625,0,-3.125,-6,-7.875,-8,-5.625,0};      double mult, sum = 0;      for(j=0;j<n-1;j++)          {              for(i=n-1;i>j;i--)                  f[i]=(f[i]-f[i-1])/(x[i]-x[i-j-1]);          }          for(i=n-1;i>=0;i--)          {              mult=1;              for(j=0;j<i;j++)                  mult\*=(a-x[j]);              mult\*=f[j];              sum+=mult;          }          return sum;  }  int main()  {      int i = 0;      double total,ganjil=0, genap=0, h = 0.01,a=-1,result;      double x[] = {-1,-0.5,0,0.5,1,1.5,2,2.5,3,3.5,4}; double f[] = {0,3.375,4,2.625,0,-3.125,-6,-7.875,-8,-5.625,0};      cout <<"TURUNAN PADA F(X)= 0.75"<<endl;      cout<<"-----------------------------"<<endl;      cout<<"f(0.75)"<<endl;      cout<<"|  x\t|  y\t |"<<endl;      cout<<"-------------------"<<endl;      for(int i=0;i<11;i++)          cout<<"|  "<<x[i]<<"\t|"<<f[i]<<"\t|"<<endl;      cout << "SELISIH MAJU : ";      result = ((interpolasi(0.75+h))-interpolasi(0.75))/h;      cout<<result;      cout<<endl;      cout << "SELISIH MUNDUR :";      result = (interpolasi(0.75)-interpolasi(0.75-h))/h;      cout<<result;      cout<<endl;      cout << "SELISIH PUSAT : ";      result = (interpolasi(0.75+h)-(interpolasi(0.75-h)))/(h+h);      cout<<result;      while(a<2){          if(i == 0)              total = interpolasi(a);          else{              if(i%2!=0)                  ganjil += interpolasi(a);              else                  genap += interpolasi(a);          }          i++;          a+=h;      }      cout<<"\nINTEGRAL DARI X = -1 SAMPAI X = 2 ADALAH : ";      total = total + (4\*ganjil) + (2\*genap) + interpolasi(a);      total = h/3 \* total;      cout<<total;  } |

1. **HASIL UJI COBA DAN ANALISA SINGKAT**

|  |
| --- |
|  |

**Gambar 4.1** Hasil *run* program soal

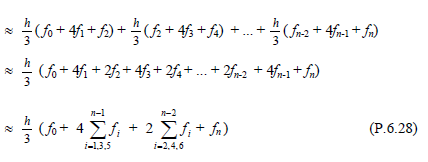
Analisis *:*

Percobaan yang telah dilakukan bertujuan untuk mendapatkan nilai turunan pertama numerik dengan menggunaakn selisih maju, mundur dan terpusat pada titik x = 0.75 dengan serta untuk mendapatkan nilai integral menggunakan metode NC 3/8 dari range -1 hingga 2 h = 0.01.

Untuk mencari nilai integral maupun turunan maka terlebih dahulu mendefinisikan persamaan fungsi menggunakan metode interpolasi Beda Terbagi Newton (BTN). Setelah fungsi didefinisikan maka berdasarkan nilai x dapat dicari fx1, fx2, fx-1, fx-2. Setelah diketahui nilai dari persamaan tersebut maka nilai dari turunan dapat dihitung menggunakan masing-masing rumus turunan.

Berdasarkan program yang dijalankan , turunan pertama menggunakan selisih maju hasilnya f’(0.75) = -5.3299, turunan pertama menggunakan selisih mundur hasilnya f’(0.75) = -5.2949, dan selisih pusat menghasilkan f’(0.75) = -5.3124. Nilai selisih mundur menghasilkan nilai terkecil dari ketiga hasil yang didapatkan, sedangkan nilai selisih maju menghasilkan nilai terbesar. Nilai selisih pusat mendapatkan hasil ditengah-tengah dari kedua hasil sebelumnya dikarenakan membandingkan antara perhitungan sellisih maju maupun selisih mundur sehingga nilai yang dihasilkan dapat lebih teliti daripada selisih maju maupun mundur.

Untuk menentukan integral dari titik -1 s/d 2 dengan h = 0.01 menggunakan rumus Newton – Cotes 1/3, maka terlebih dahulu menentikan hasil fx dari setiap nilai x yang ada kemudian dari fx yang didapatkan dimasukkan ke dalam rumus :

****

Berdasarkan perhitungan program yang dilakukan maka nilai integral dari titik -1 hingga 2 adalah 2.25.

1. **KESIMPULAN**
2. Pencarian turunan pertama dari suatu sebaran data dapat dilakukan dengan selisih pusat, selisih maju maupun selisih mundur dengan terlebih dahulu mendefinisikan persamaan dari sebaran data yang ada melalui interpolasi Beda Terbagi Newton lalu dimasukkan ke dalam fungsi pendekatan sehingga didapatkan hasil turunan dari nilai balik fungsi tersebut. Dengan mengetahui konsep tersebut kemudian dapat memudahkan dalam pengimplementasian beserta pembelajaran saat mengimplementasikannya ke dalam sebuah program C++.
3. Nilai integral suatu sebaran data dapat dihitung menggunakan berbagai metode salah satunya yaitu Newton-Cotes (NC Simpson 1/3). Dimana pada metode tersebut dilakukan penjumlahan semua nilai terhadap f(x) berdasarkan f(x) ganjil maupun f(x) genap terkecuali pada f(x0) maupun f(xn) lalu dihitung menggunakan rumus Simpson 1/3 yang kemudian diimplementasikan ke program C++.
4. **REFERENSI**

<https://en.m.wikipedia.org/wiki/Numerical_integration>

<http://data.fmipa.unand.ac.id/matematika/file_bahankuliah/5.%20Turunan%20Numerik.pdf>

<http://informatika.stei.itb.ac.id/~rinaldi.munir/MetNum/2010-2011/Turunan%20Numerik.pdf>

<https://id.scribd.com/doc/55739479/Turunan-Numerik>