LAPORAN OBSERVASI

Tugas Pemrograman 01

CTI-2G3 Sistem Cerdas

Oleh:

Alfan Cahyo Wicaksono (1303184065)

Deni Saputra Hermawan (1303180074)

Mohammad Rizqi Akmaludin (1303184031)

PROGRAM STUDI S1 TEKNOLOGI INFORMASI

FAKULTAS INFORMATIKA

UNIVERSITAS TELKOM

2021/2022

DESKRIPSI

Dalam mengimplementasikan algoritma genetika, terdapat beberapa bagian, yaitu inisiasi populasi yang terdiri dari 10 buah gen menggunakan generate kromosom, evaluasi dengan menghitung fitness dan error, crossover dengan menggabungkan antara 2 parent yang telah digenerate, mutasi menggunakan random mutation dengan range 0 hingga panjang kromosom, seleksi dan regenerasi, dan membuat populasi baru dari hasil seleksi dan regenerasi.

DATASETS

Datasets diambil dari STOCK PRICE DATA
Top 10 leading IT Stock price data from NSE India stock exchange. Below are the 10 stocks listed in dataset :

1 Tata Consultancy Services- TCS
2 Infosys- INFY
3 Wipro- WIPRO
4 HCL Tech- HCLTECH
5 Tech Mahindra- TECHM
6 Larsen & Toubro Infotech- LTI
7 MindTree- MINDTREE
8 Oracle Financial Services Software- OFSS
9 Mphasis- MPHASIS
10 L&T Technology Services- LTTS

Justifikasi : Kami mengambil data prev closenya saja dari tanggal 5 maret 2019 hingga 8 agustus 2019 dengan total row 110.
Sumber : Kaggle

IMPLEMENTASI FUNGSI

1. Library

```
1   from random import choices, randint, uniform
2   from typing import List
3   import numpy as np
4   import pandas as pd
5   from math import sqrt
```

Library yang digunakan pada program ini ada beberapa diantaranya adalah Random, pandas, numpy, math. Setiap library mempunyai fungsi masing-masing seperti pengacakan angka, pemrosesan atau operasi bilangan dan pengaksesan file dataset.

2. Pengenalan list dan ukuran dari populasi serta gerasi yang diinginkan

```
7   Nilai_saham = List[int]
8   Kromosom = List[int]
9   Populasi = List[Kromosom]
10  max_pop = 200 # DAPAT DIRUBAH SESUAI DENGAN MAX POPULASI YANG DIINGINKAN
11  max_generation = 1000
12
```

Pada program terdapat beberapa variabel yang digunakan. Diantaranya adalah yang sudah di tuliskan pada gambar diatas, dimana tiap variabel akan menampung data berupa list ataupun integer yang akan diolah pada fungsi atau prosedur yang ada.

3. Generate Kromosom

```
14  def generate_kromosom() -> Kromosom:
15      # melakukan generate nilai dari a1 sampai a10
16      return [uniform(-1, 1) for i in range(11)]
17
```

Fungsi def generate_kromosom digunakan untuk melakukan generate nilai dari a1 hingga ke a10. Nilai yang digenerate sesuai dengan yang ada pada dataset yang digunakan.

4. Harga Saham

```
19  def harga_saham(konstanta: Kromosom, nilai: Nilai_saham):
20      # dari soal f(x) = a0 + a1.y1 + a2.y2 + a3.y3 + .... + a10.y10
21      # a = konstanta, y = nilai
22      nilai = np.insert(nilai, 0, 1)
23      return sum(np.multiply(konstanta, nilai))
```

Fungsi def harga_saham digunakan untuk melakukan operasi hingga menghasilkan nilai f(x). Rumus yang digunakan yaitu f(x)=a0 + a1.y1 + a2.y2 + a3.y3 + ... + a10.y10.

5. Hitung Fitness

```python
def hitung_fitness(kromosom: Kromosom, saham: Nilai_saham) -> float:
    # rumus untuk nilai error (Mean Squared Error)
    squared_error = 0
    for i in range(10):
        y = harga_saham(kromosom, saham[i+1:i+11])
        error = saham[i] - y
        squared_error += error ** 2
    mse = sqrt(squared_error/10)
    return 1/(0.00000000000000000000000000000000000000000001+mse)
```

Fungsi def hitung_fitness digunakan untuk menghitung nilai fitness dengan menghitung prediksi harga saham, serta menghitung nilai error.

6. Populasi Awal

```python
def populasi_awal() -> Populasi:
    # untuk melakukan generate populasi awal dengan range hingga makpop (sesuai yang diinginkan)
    return [generate_kromosom() for i in range(max_pop)]

```

Fungsi def populasi_awal digunakan untuk melakukan generate populasi dimana populasi akan digenerate hingga nilai maksimal pop yang diberikan. Pada kasus ini maksimal popnya adalah 100.

7. Regenerasi Populasi

```python
def regen_pop(populasi: Populasi, parent: Populasi, pc: int) -> Populasi:
    # menampilkan regenerate populasi berdasarkan mutasi yang dilakukan
    populasi = populasi[:len(populasi)-pc]
    populasi += crossover(parent[0], parent[1], pc)
    return [mutasi(kromosom) for kromosom in populasi]

```

Fungsi def regen_pop digunakan untuk menampilkan hasil regenerate populasi berdasarkan mutasi yang sudah dilakukan.

8. Parent Selection

```python
def parent_selection(populasi: Populasi, saham: Nilai_saham, fit: Populasi) -> Populasi:
    # memilih parent berdasarkan populasi
    return choices(
        populasi,
        weights=fit,
        k=2
    )
```

Fungsi def parent_selection digunakan untuk melakukan proses pemilihan parent berdasarkan populasi yang ada.

9. Crossover

```python
def crossover(parentA: Kromosom, parentB: Kromosom, pc: int) -> Populasi:
    # mengawinkan parent
    offspring = []
    for x in range(0, pc-1, 2):
        i = randint(1, 10)
        offspring += [parentA[0:i] + parentB[i:], parentB[0:i] + parentA[i:]]
    if pc % 2:
        i = randint(1, 10)
        offspring += choices([parentA[0:i] + parentB[i:],
                              parentB[0:i] + parentA[i:]], k=1)
    return offspring
```

Fungsi def crossover digunakan untuk menggabungkan parent berdasarkan 2 kromosom

10. Mutasi

```python
def mutasi(kromosom: Kromosom) -> Kromosom:
    for i in range(0, len(kromosom)):
        if np.random.random_sample() < pm:
            kromosom[i] = uniform(-1, 1)
    return kromosom
```

Fungsi def mutasi Menghasilkan individu atau populasi baru dengan gen yang berbeda dari hasil random sample. Populasi atau parent dipilih secara random. Banyaknya individu yang dihasilkan bergantung pada nilai uniform.

11. Fungsi Main

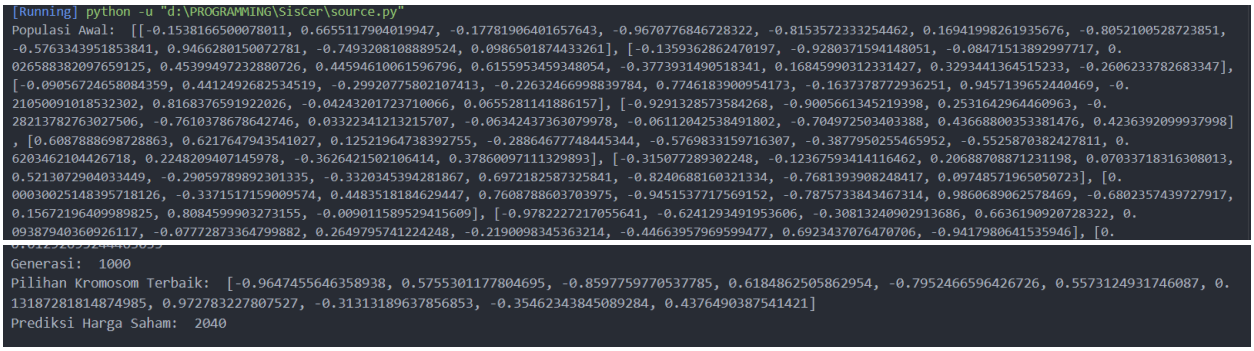
```python
# fungsi main
pop = populasi_awal()
print('Populasi Awal: ', pop)

# Probabilitas operasi genetik (Pc dan Pm)
pm = 1/(len(pop)*len(pop[0]))  # probabilitas mutasi = 1 / banyak gen
pc = round(0.4 * max_pop)

gen = 0
# membaca datasets yang ada pada excel
dataset = pd.read_excel('datasets.xlsx', usecols='B')
awal = 20
saham = dataset.values[awal:awal+21]
harga = saham[0]
nilai = saham[1:11]
error = 1
while gen < max_generation:
    gen += 1
    fit = [hitung_fitness(kromosom, saham) for kromosom in pop]
    pop = [x for _, x in sorted(zip(fit, pop), reverse=True)]
    fit = sorted(fit, reverse=True)
    print(fit[0])
    if (fit[0] > error):
        break

    parent = parent_selection(pop, saham, fit)

    pop = regen_pop(pop, parent, pc)

fit = [hitung_fitness(kromosom, saham) for kromosom in pop]
pop = [x for _, x in sorted(zip(fit, pop), reverse=True)]

print('Generasi: ', gen)
print('Pilihan Kromosom Terbaik: ', pop[0])
print('Prediksi Harga Saham: ', round(harga_saham(pop[0], saham[:10])))
```

Implementasi seluruh fungsi dengan pemanggilan dan output hasil yang diharapkan yakni berupa prediksi harga saham pada akhir baris kode.

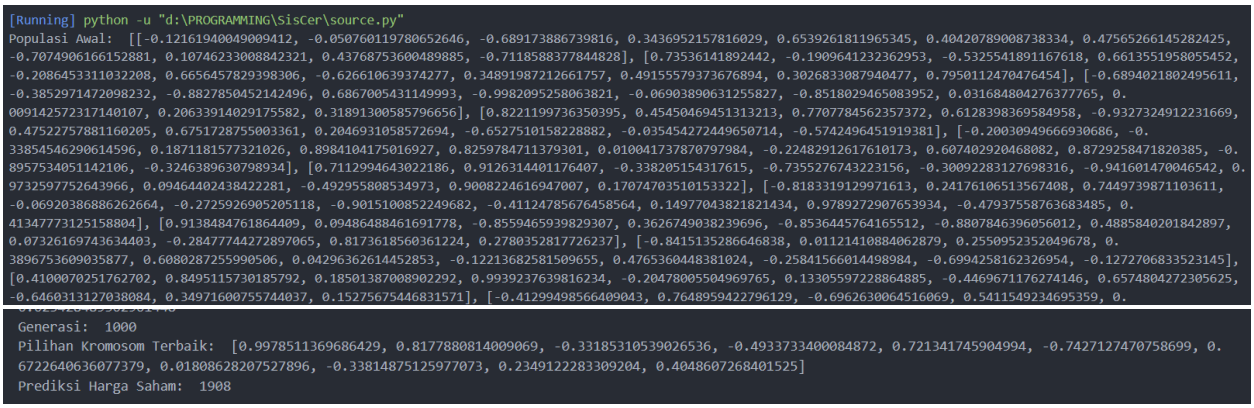
HASIL PENGUJIAN

Setiap hasil uji coba command window. Dimana akan menampilkan populasi awal, regenerate populasi, Hasil pemilihan Kromosom terbaik dan Prediksi saham.

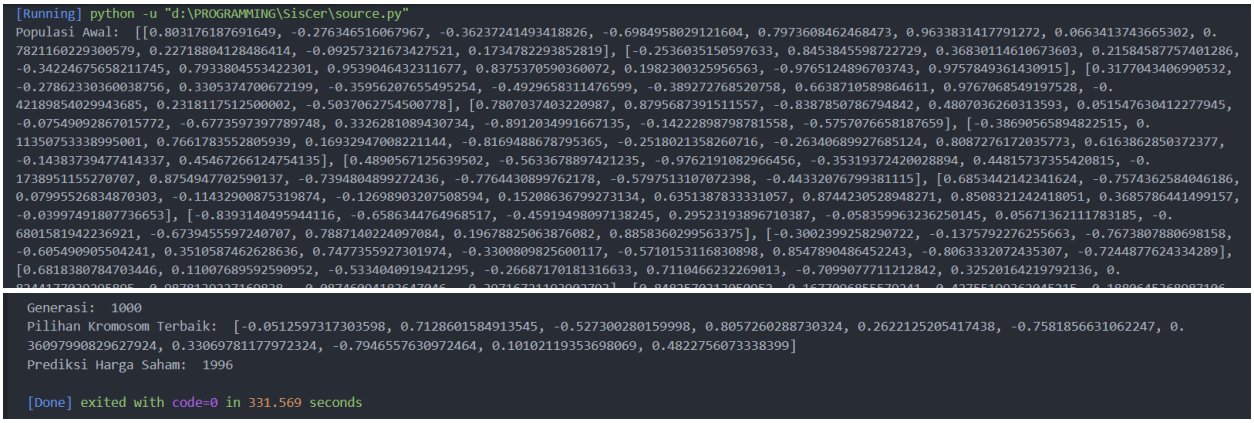
1. Uji Coba Populasi
   Ukuran populasi yang diujikan adalah pada interval [200, 1000] dengan kelipatan 400. Kombinasi kromosom dan mutasi yang digunakan adalah -1:1.
   a. Populasi 200



   b. Populasi 600



   c. Populasi 1000



2. Uji Coba rentang uniform dari kromosom dan mutasi
   Ukuran populasi yang diujikan adalah 100. Kombinasi kromosom dan mutasi yang digunakan adalah -1:1, -10:10, -100:100
   a. Uniform -1:1

```
[Running] python -u "d:\PROGRAMMING\SisCer\source.py"
Populasi Awal:  [[-0.17306724020544562, 0.21781616119637093, -0.798613052949223, 0.19568993795906708, -0.22301539947123117, 0.72325540149852, -0.18468178324979734,
-0.9540664754908017, -0.06736371268390662, 0.2295494661337203, -0.29515749378431844], [0.9365419513671178, 0.030857673670701358, -0.3941025697759588, 0.
7745771553503422, -0.6346211029478035, 0.2238115833535672, -0.9401742146512384, -0.727788506311312, -0.3122488401199288, -0.1202755016292183, -0.47761937354703354],
[-0.02807376495763303, -0.40380159664495663, -0.907262903979609, -0.5607024195545944, 0.4575315117598473, 0.5896353485838577, 0.22831752215586154, 0.
9886702875985101, -0.06403624650244177, -0.9100000315319239, 0.5208864040611285], [-0.3619233379543261, -0.9066606880547308, 0.00633092278728342 2, -0.
17676187092590512, -0.8748777414634044, -0.24257522804674592, 0.4011998404270176, -0.07233409296570703, 0.5932929133708533, -0.8599124833389495, -0.9025598646355233]
, [-0.449146616257345, -0.2540114423299076, 0.2665281456140265, 0.7094060069506354, 0.4213115130604539, -0.7220971901636626, 0.8093670674144577, -0.9945810306579497,
0.9491462158793105, 0.2143129064090883, -0.5863904044591213], [-0.9841718067808363, -0.8753614312820863, -0.3997332771135047, -0.7996738089443471, 0.
8754148366714183, -0.0902813459467362, 0.14758106385033565, -0.23637109875712858, 0.9160824490067307, 0.4044177720477662, -0.37252603301134113], [0.7611737220803199,
0.0835233305615819, 0.9435772907755584, 0.4225284798615845, -0.7334900255425993, 0.6787426030720669, -0.2972133383317150 4, 0.5515859013909163, -0.09228027626963664,
-0.6783505143709121, 0.9647075842770478], [0.5670401888069108, -0.5531351120268571, 0.22316337697032496, -0.5131301781973918, -0.15878820957506434, 0.
47274804204201915, 0.16056111710830723, -0.7100529830055864, 0.16017479575662996, -0.16882499735425216, -0.06876396600751344], [0.2223405267995826, 0.
6039233058483384, 0.8860857030843072, -0.9358986543503685, 0.9309031504820338, 0.5945839966274462, 0.10472264482226334, 0.7206765189042146, 0.6648467370426103, -0.
6916327419160273, -0.7631511008103127], [0.8602465215314616, -0.9034920234125148, 0.5744829405318137, -0.5182570174966121, -0.4043299639820561, -0.43911333100071803,
-0.8952437474443931, -0.8324912501117285, 0.55388170284364, 0.3925754918120963, -0.39628386792065196], [0.928216818208528, 0.8258387880400186, -0.27913983979011, 0.
  0.018688498367745658
  Generasi:  1000
  Pilihan Kromosom Terbaik:  [0.4005300100836837, 0.2698038136184342, 0.6482341312064974, -0.5557596376282263, -0.445904869862884, 0.2755786771766635, -0.
  2942623413635548, 0.3888235027404743, -0.46717532455618294, 0.4030296444671393, 0.7259842149944651]
  Prediksi Harga Saham:  1934
```

b. Uniform -10:10

```
[Running] python -u "d:\PROGRAMMING\SisCer\source.py"
Populasi Awal:  [[9.890430490583213, 7.996634901193396, -4.133099154963373, 1.116507854035822, 9.496068152313597, -5.4822814974823935, 1.9786673491738842, -4.
234270107891998, 8.366850341043843, -9.6610512182011, 3.7700905459843383], [5.279480436111896, 8.717516391969678, -8.660295331575076, 0.3521673481889227, 8.
471005697023472, -0.5918241595926315, 7.229354792960294, 1.7933131382119853, -3.48317037933705, -1.389034295392996, -4.888575903041259], [-0.9494494758651655, 8.
809445922612696, 6.574422388648994, -8.26799778659427, -6.773780672423175, -8.605770385055498, 1.5329899430479426, -3.541156712410743, 3.715484820635231, 2.
528148242826834, -9.201067308316473], [-3.564269283561659, -0.6592521122160626, -9.444488515196728, 3.7607926090425003, -6.490942194141802, 4.239349457317314, -1.
3920244170346585, 8.730423344165452, 1.877064413864007, 9.795842086613249, -7.406209587140024], [3.4511517169160673, -6.38954664896815, -0.2624480750449276, 5.
176290464462966, -2.008671968473972, -9.423790275483466, -7.328201584088854, -2.28047499163387, -6.542837411121645, -9.226039711548095, -3.681854443542063], [-1.
3089832821505194, -4.835011980234 56, 8.339852881570089, -1.3960198748282195, -6.1926958472143845, -4.173641530062615, 8.580006644334567, -0.5370668777787575, -6.
288190451443281, -4.065832093978758, 2.2875206818256917], [0.4247784226765621, -6.475618548878299, 0.2721371719519041, 6.296809540708001, 8.525989089143547, 5.
969482449725582, 8.364290100131104, -6.752298396504665, 7.715043186563765, -0.5559976278816787, 7.368445661804937], [-6.954990644552184, 1.0063879512765066, 3.
2652335462676625, 9.971857181424696, -6.830633174845 63, -8.524880081471 62, 6.978332852784934, -3.796066398070213, 0.33407786727521 94, 0.08285763165998894, 3.
529378730867057], [-4.849791017978293, -9.805847200556176, 5.964765407538447, -7.114797468690412, 0.4014302828279437, 7.832728397567955, -0.21375417631831262, -2.
7245248175487546, 0.5162952740186437, -1.0420713198769214, -2.9440312543797553], [4.228214365049418, 0.3555561814676338, 5.782428507292426, -3.7621627163135596, -5.
```

```
  Generasi:  1000
  Pilihan Kromosom Terbaik:  [-9.487999904431197, -7.037127424653818, -4.238046041479874, 1.873907053954051, -5.683258909644666, 7.851963768685952, 1.6508901544805603,
  9.159489376540055, -5.613859782648336, -1.4460334286074517, 4.150597025747317]
  Prediksi Harga Saham:  1975
```

c. Uniform -100:100

```
[Running] python -u "d:\PROGRAMMING\SisCer\source.py"
Populasi Awal:  [[-9.079652190254478, 82.7406643355979, -56.57846049335298, -71.785528451101, -70.82059356471663, 45.42523517650153, 84.13706253106119, -24.
619891838660223, -40.27590466308788, -63.44661923965709, 88.23536650598103], [-97.25594331182641, 36.635368994359254, -53.452723972554764, 31.907933695695 42, 31.
00048297190088, -29.151185634045177, 29.864392331599618, 76.09436045991106, -51.886156828874654, -65.72286277622963, 86.89199481803436], [-38.79697644801525, 95.
04208703368639, -47.72471343913129, -33.09581539743975, 54.43990247401197, 58.76449534864204, 7.40060693418971, 40.18889455576132, 66.51613235592436, 22.
744017921862365, -98.11189623180825], [-65.5438806945414, -66.87604432448977, -22.71884296464357, -90.3700552355013, -47.9574171844413 54, 8.098095869505784, -10.
588911616176432, -86.02154268628547, -21.800589159929444, -98.26454250991554, 87.76592144013472], [-70.49188494728216, 27.918110960027406, -4.047904213806504, -65.
06624903812454, 84.92334527178642, -10.376483619467749, -8.87807647287768, -74.92762463173632, 42.752811317310346, -85.90124894314985, 36.543582424619075], [-40.
  Generasi:  1000
  Pilihan Kromosom Terbaik:  [92.2454604094024, -29.45222783398674, 8.291264593279507, -36.6889469747302, -35.401551406687275, 60.23922435044784, 64.93837282932495,
  -35.23370092439413, -41.32004131169662, 34.643201256233766, 8.813525156129359]
  Prediksi Harga Saham:  -2527
```

HASIL REVISI

Problem : Tidak menampilkan secara selesai prediksi 50 data saham baru

Solving : Berdasarkan revisi yang dibutuhkan yakni nilai prediksi atau forecast saham dalam 50 hari. Algoritma baru ini akan mengecek best kromosom dahulu selama 50 hari. Dengan rule 10 hari sebelumnya sesuai fungsi pada soal. Pada output forecast saham dapat dilihat prediksi dengan format yakni [Hari ke berapa, harga prediksi, harga sebenarnya, selisih harga]. Untuk hasilnya sendiri dapat dilihat masih terdapat selisih yang cukup besar diatas 200, untuk itu masih belum akurat.

REFERENSI

1. Prediction of Stock Market Index Using Genetic Algorithm R. Lakshman Naik1*, D. Ramesh1 , B. Manjula1 , Dr. A. Govardhan2
2. Implementasi Genetic Algorithm dalam Model ARIMA untuk Memprediksi Observasi Time Series Rangga Arya Pamungkas #1 , Indwiarti *2 , Aniq Atiqi Rohmawati #3
3. PREDIKSI HARGA SAHAM BERDASARKAN DATA HISTORIS MENGGUNAKAN MODEL REGRESI YANG DIBANGUN DENGAN ALGORITMA GENETIKA Asyrofa Rahmi, Wayan Firdaus Mahmudy, Budi Darma Setiawan
4. https://medium.com/analytics-vidhya/stock-price-prediction-using-genetic-algorithms-65c64b002d9
5. https://github.com/Shidqiet/stock-price-prediction-genetic-algorithm
6. https://www.kaggle.com/sunnykusawa/stock-price-data
7. https://github.com/antoneklund/Stock-prediction-GA/blob/master/equationGA.py
8. https://www.youtube.com/watch?v=nhT56blfRpE
9. https://gist.github.com/NoblesseCoder/d35e19e441ed014ea69ff800be28e748
10. http://www.worldscientificnews.com/wp-content/uploads/2015/07/WSN-19-2015-148-167.pdf

LINK VIDEO

https://drive.google.com/file/d/1JuRzZq_yyEx8PrOMxsYCe_pYgoILbrg6/view?usp=sharing