Laporan Tugas Kecil 2 IF2211 Strategi Algoritma
Semester II Tahun 2022/2023

Mencari Pasangan Titik Terdekat 3D dengan *Algoritma Divide and Conquer*

Disusun oleh:
Azmi Hasna Zahrani
13521006
K03

TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG

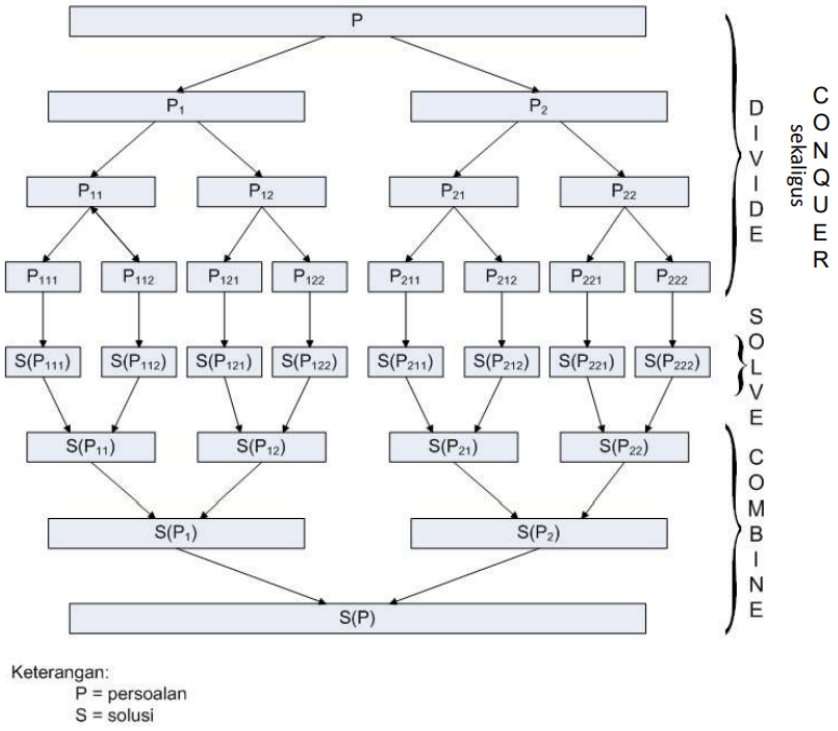
2023

DAFTAR ISI

BAB I
LATAR BELAKANG

1.1 Algoritma *Divide and Conquer*

Divide and Conquer merupakan algoritma yang diambil dari salah satu strategi militer bernama *Devide et Impera* yang berarti memecah dan menguasai. Seperti namanya, algoritma ini dijalankan dengan beberapa langkah, yaitu *divide, conquer,* dan *combine. Divide* berarti membagi persoalan besar ke dalam sub-persoalan yang mirip dengan persoalan asli tetapi berukuran lebih kecil. *Conquer* berarti menyelesaikan masing-masing sub-persoalan dengan kondisi apabila persoalan masih berukuran besar maka persoalan tersebut diselesaikan secara rekursif dan apabila persoalan sudah berukuran kecil maka persoalan tersebut diselesaikan secara langsung. *Combine* berarti menggabungkan solusi masing-masing sub-persoalan yang telah didapat dari langkah sebelumnya sehingga membentuk solusi dari persoalan secara menyeluruh. Algoritma *Divide and Conquer* dapat digunakan untuk menyelesaikan beberapa persoalan seperti pencarian nilai minimum dan maksimum, menghitung perpangkatan, *sorting,* mencari titik terdekat, perkalian matriks, perkalian bilangan bulat besar, perkalian dua buah polinom, dan lain sebagainya.



Gambar 1.1 Bagan *Divide and Conquer*
Sumber: Bahan Kuliah IF2211 Strategi Algoritma: Algoritma Divide and Conquer Bagian 1

1.2 Penggunaan Algoritma *Divide and Conquer* dalam Menyelesaikan Masalah Pencarian Titik Terdekat

Pencarian titik terdekat pada bidang 3 dimensi dapat diselesaikan dengan menggunakan pendekatan algoritma *Divide and Conquer*. Misal pada sebuah bidang 3 dimensi terdapat n buah titik yang memiliki koordinat x, y, dan z. Jarak terdekat dari dua buah titik dapat dihitung menggunakan rumus Euclidean $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 + ...}$ begitu pula untuk dimensi 4, 5, dst. Menggunakan algoritma *Divide and Conquer,* bidang dimensi yang memuat titik-titik akan dibagi menjadi dua sama besar (memiliki titik yang sama banyak) dan operasi perhitungan akan dilakukan secara rekursif apabila di kedua bagian masih didapati banyak titik. Ketika titik yang tersisa pada hasil pembagian bidang kurang dari atau sama dengan tiga, maka akan dilakukan perhitungan euclidean secara langsung. Titik-titik tersebut akan divisualisasikan dalam bidang 2 dimensi dan 3 dimensi.

BAB II
SOURCE CODE PROGRAM DALAM BAHASA PYTHON

Dalam program yang penulis buat, terdapat dua file program yaitu function.py dan main.py

2.1 function.py

```
import numpy as np
import random
import matplotlib.pyplot as plt

def createPoint (num, dim):
    #createPoint using randomize
    #num as number of points
    #dim as dimension of points
    #return a list of num points in dim dimensions
    points = np.zeros((num, dim))
    for i in range(num):
        for j in range(dim):
            points[i][j] = random.randint(1, 100)
    return points

def euclideanDistance (point1, point2, count):
    count = count + 1
    distance = 0
    for i in range(len(point1)):
        distance += (point1[i] - point2[i])*(point1[i]- point2[i])
    return np.sqrt(distance), count

def closestBF(points, count):
    #brute force
    distance = euclideanDistance(points[0], points[1], count)
    point1 = points[0]
    point2 = points[1]
    for i in range(len(points)):
        for j in range (1+i, len(points)):
            brute = euclideanDistance(points[i], points[j], count)
            count = count + 1
            if (brute < distance):
                distance = brute
                point1 = points[i]
                point2 = points[j]
```

```
    return distance, point1, point2, count


def closestDnC (points, count):
    if (len(points) <= 3):
        return closestBF(points, count)

    else :
        #slice matrix into 2 parts
        half = len(points)//2
        disA, pointA1, pointA2, count= closestDnC(points[:int(half)],
count)
        disB, pointB1, pointB2, count = closestDnC(points[int(half):],
count)
        if disA < disB:
            distance = disA
            point1 = pointA1
            point2 = pointA2
        else:
            distance = disB
            point1 = pointB1
            point2 = pointB2
        #sStrip
        middle = points[int(half)][0]
        sStrip = []
        for i in range(len(points)):
            if (abs(points[i][0] - middle) < distance).any():
                sStrip.append(points[i])
        #compare sStrip

        for i in range(len(sStrip)):
            for j in range(1+i, len(sStrip)):
                if (abs(sStrip[i][1] - sStrip[j][1]) < distance).any():
                    strip = euclideanDistance(sStrip[i], sStrip[j], count)
                    count = count + 1
                    if strip < distance:
                        distance = strip
                        point1 = sStrip[i]
                        point2 = sStrip[j]
    return distance, point1, point2, count
```

```python
def plot(points, point1, point2):
    # plot the points
    # d is the dimension of the points
    dim = len(points[0])
    if dim == 2:
        x = points[:, 0]
        y = points[:, 1]
        plt.scatter(x, y, color='black', alpha=0.5, s=10, marker='o')
        plt.scatter(point1[0], point1[1], color='red')
        plt.scatter(point2[0], point2[1], color='red')
        plt.show()
    elif dim == 3:
        x = points[:, 0]
        y = points[:, 1]
        z = points[:, 2]
        fig = plt.figure()
        ax = fig.add_subplot(111, projection='3d')
        ax.scatter(x, y, z, color='black' , alpha=0.5, s=10 , marker='o')
        ax.scatter(point1[0], point1[1], point1[2], color='red')
        ax.scatter(point2[0], point2[1], point2[2], color='red')
        plt.show()
    else:
        print('Cannot plot the points in dim dimensions')
```

2.2 main.py

```python
from function import *
import time

print ("""
    __  _        ___    _____  ___  _____ ______      ____   ____  ____
____
   /  | | |     /   \ / ____/ /  _]/ ___/|      |    |    \ /    ||    ||
\
  /  /  | |    |     (   \_  /  [_(   \_ |      |    |  o  )  o  | |  | |
D  )
 /  /   | |___ |  O  |\__  ||    _]\__  ||_|  |_|    |   _/|     | |  | |
/
/   \_  |     ||     |/  \ ||   [_ /  \ |  |  |      |  |  |  _  | |  | |
\
```

```
\     | |     ||     |\    ||     |\    |  |  |      |  |  |  |  | |  | |
.  \\
 \____| |_____| \___/  \___||_____| \___|  |__|      |__|
|__|__||____||__|\_|

""")

print ("START? (Y/N)")
start = input()
if start == "Y" or start == "y":
    print('Enter the number of dimensions:')
    dim = int(input())
    print('Enter the number of points:')
    num = int(input())
    count = 0
    points = createPoint(num, dim)
    print(points)
    print ('------------------------------------')
    print('Divide and Conquer:')
    startDnC = time.time()
    distance, point1, point2, count = closestDnC(points, count)
    endDnC = time.time()
    print('The closest distance is: ', distance)
    print('The closest points are: ', point1, ' and ', point2)
    print('The number of Euclidean calls: ', count)
    print('The time taken: ', round((endDnC - startDnC)*1000,2), "ms")
    print ('------------------------------------')
    print('Brute Force:')
    startBF = time.time()
    distance, point1, point2, count = closestBF(points, count)
    endBF = time.time()
    print('The closest distance is: ', distance)
    print('The closest points are: ', point1, ' and ', point2)
    print('The number of Euclidean calls: ', count)
    print('The time taken: ', round((endBF - startBF)*1000,2), "ms")
    print ('------------------------------------')
    plot(points, point1, point2)
elif start == "N" or start == "n":
    print('Goodbye')
else:
```

```
    print('Invalid input')
    print('Goodbye')
```

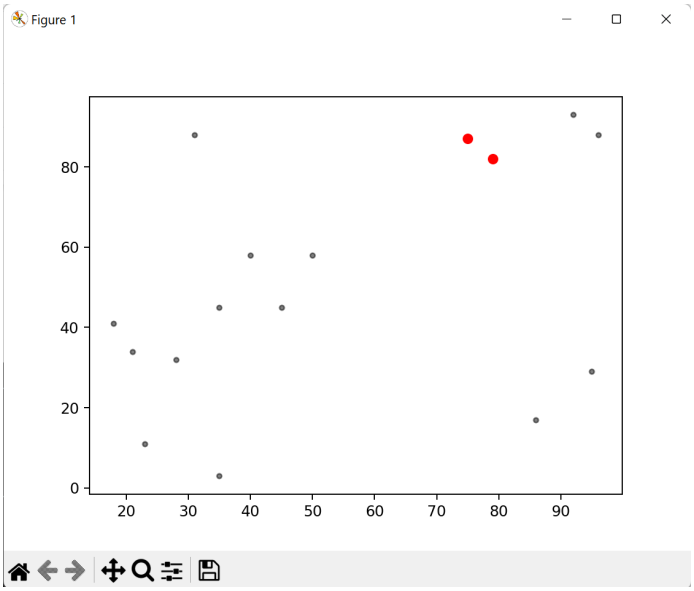
BAB III
HASIL PENGUJIAN

3.1 Hasil Pengujian 2 Dimensi

3.1.1 Pengujian 16 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
 [21. 34.]
 [92. 93.]
 [31. 88.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (6.4031242374328485, 1)
The closest points are:  [75. 87.]  and  [79. 82.]
The number of Euclidean calls:  15
The time taken:  1.25 ms
-----------------------------------
Brute Force:
The closest distance is:  (6.4031242374328485, 16)
The closest points are:  [75. 87.]  and  [79. 82.]
The number of Euclidean calls:  135
The time taken:  1.0 ms
-----------------------------------
```

Gambar 3.1.1.1 Hasil Pengujian 16 Titik pada Bidang 2 Dimensi



Gambar 3.1.1.2 Visualisasi Hasil Pengujian 16 Titik pada Bidang 2 Dimensi

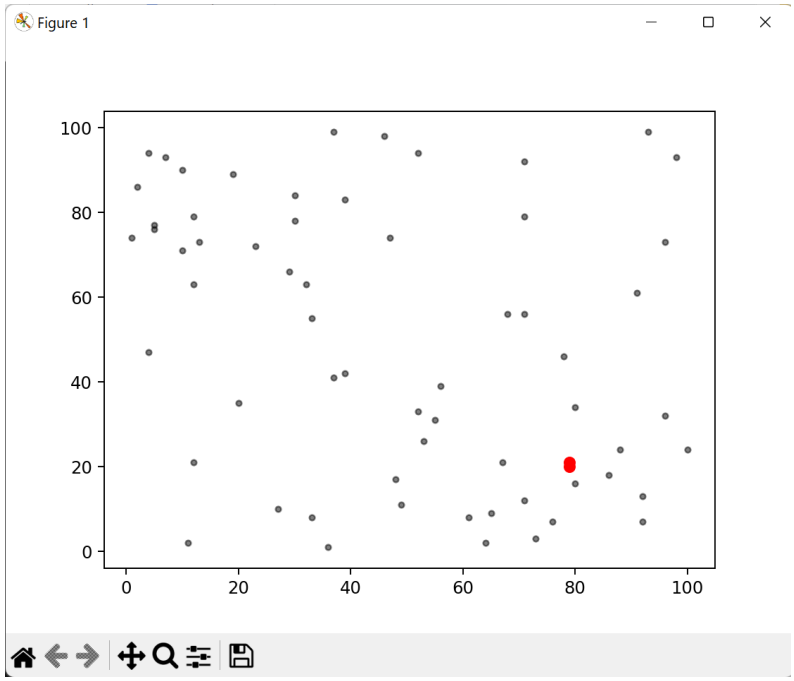
3.1.2 Pengujian 64 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
2
Enter the number of points:
64
[[ 12.  79.]
 [ 96.  73.]
 [ 23.  72.]
 [ 33.   8.]
 [ 73.   3.]
 [ 12.  21.]
 [  4.  47.]
 [ 79.  20.]
 [ 71.  79.]
 [ 80.  16.]
 [ 46.  98.]
 [  5.  76.]
 [ 27.  10.]
 [ 65.   9.]
 [ 19.  89.]
 [ 37.  99.]
 [ 49.  11.]
 [  7.  93.]
 [ 76.   7.]
 [ 36.   1.]
 [ 79.  21.]
 [ 80.  34.]
 [ 93.  99.]
 [ 29.  66.]
 [ 61.   8.]
 [ 56.  39.]
 [ 33.  55.]
 [100.  24.]
 [ 52.  33.]
```

Gambar 3.1.2.1 Hasil Pengujian 64 Titik pada Bidang 2 Dimensi

```
 [  2.  86.]
 [  5.  77.]
 [ 10.  90.]
 [ 86.  18.]
 [ 10.  71.]
 [ 30.  78.]
 [ 71.  92.]
 [  1.  74.]
 [ 47.  74.]
 [ 39.  83.]
 [ 92.   7.]
 [ 98.  93.]
 [ 67.  21.]
 [ 55.  31.]
 [ 96.  32.]
 [ 39.  42.]
 [ 52.  94.]
 [ 48.  17.]
 [ 30.  84.]
 [ 11.   2.]
 [ 13.  73.]
 [ 64.   2.]
 [  4.  94.]
 [ 91.  61.]]
------------------------------------
Divide and Conquer:
The closest distance is:  (1.0, 1455)
The closest points are:  [79. 20.]  and  [79. 21.]
The number of Euclidean calls:  3038
The time taken:  44.83 ms
------------------------------------
Brute Force:
The closest distance is:  (1.0, 3471)
The closest points are:  [79. 20.]  and  [79. 21.]
The number of Euclidean calls:  5054
The time taken:  9.49 ms
------------------------------------
```

Gambar 3.1.2.2 Hasil Pengujian 64 Titik pada Bidang 2 Dimensi



Gambar 3.1.2.3 Visualisasi Hasil Pengujian 64 Titik pada Bidang 2 Dimensi

3.1.3 Pengujian 128 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
2
Enter the number of points:
128
[[ 89.  45.]
 [ 34.  52.]
 [ 94.  28.]
 [  5.  43.]
 [ 15. 100.]
 [ 36.  36.]
 [ 78.  19.]
 [ 56.  93.]
 [ 41.  98.]
 [ 31.  20.]
 [ 72.  28.]
 [100.  91.]
 [ 23.  32.]
 [ 69.  31.]
 [ 64.  76.]
 [  6.  49.]
 [ 61.  10.]
 [ 25.  94.]
 [ 10.  56.]
 [ 73.  53.]
 [ 79. 100.]
 [  3.  39.]
 [ 38.   7.]
 [ 59.  28.]
 [ 13.  71.]
 [ 64.  14.]
 [ 91.  58.]
 [ 89.  71.]
```

Gambar 3.1.3.1 Hasil Pengujian 128 Titik pada Bidang 2 Dimensi

```
[ 86.  46.]
[ 64.  16.]
[ 10.  17.]
[ 75.   7.]
[ 34.  77.]
[ 55.  31.]
[ 63.  70.]
[ 80.  65.]
[ 71.  91.]
[ 99.  99.]
[ 17.  71.]
[ 97.   3.]
[ 13.  17.]
[ 89.  20.]
[ 74.   3.]
[ 54.  85.]
[ 85.  16.]
[ 95.   7.]
[ 27.  64.]
[ 17.  93.]
[ 70.  29.]
[  6.  89.]
[ 27.  69.]
[ 42.  23.]
[ 92.  79.]
[ 98.   8.]
[ 61.  23.]
[ 82.   7.]
[ 62.  10.]
[ 90.  36.]
[ 33.  25.]
[ 37.  92.]
[ 51.  59.]
[ 80.  52.]
[ 18.   9.]
[ 82.  10.]
[ 88.  58.]
[ 70.  46.]
[ 86.  19.]
[ 54.  62.]
[ 72.  36.]
[ 77.  82.]
```

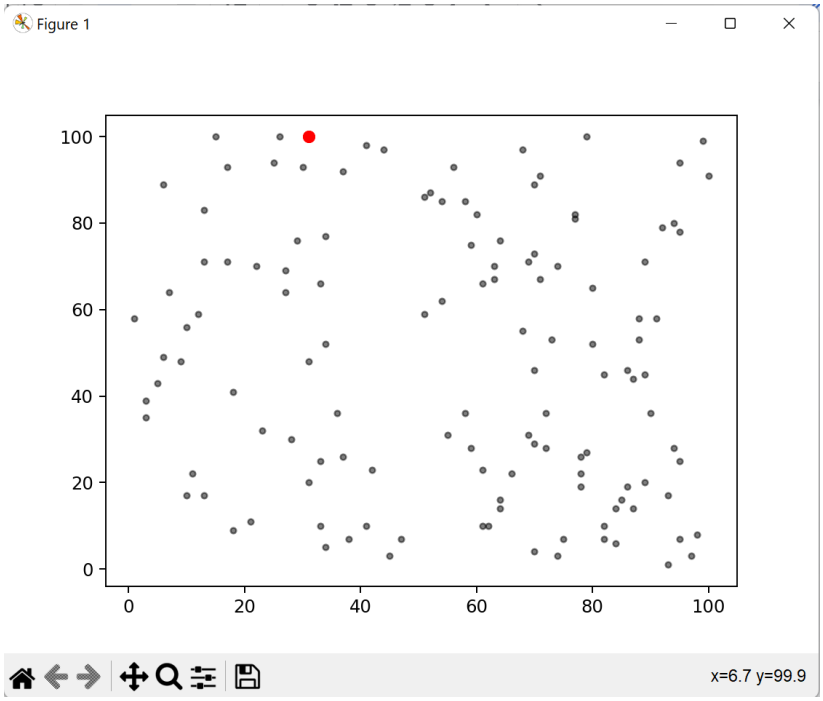
Gambar 3.1.3.2 Hasil Pengujian 128 Titik pada Bidang 2 Dimensi

```
[ 77.  82.]
[ 71.  67.]
[  1.  58.]
[ 95.  25.]
[ 12.  59.]
[ 52.  87.]
[ 31.  48.]
[ 70.  73.]
[ 33.  10.]
[ 70.  89.]
[ 87.  44.]
[ 95.  94.]
[ 34.   5.]
[ 22.  70.]
[ 87.  14.]
[ 88.  53.]
[ 47.   7.]
[ 70.   4.]
[ 84.  14.]
[ 21.  11.]
[ 84.   6.]
[ 78.  22.]
[ 94.  80.]
[ 74.  70.]
[ 58.  36.]
[ 78.  26.]
[ 79.  27.]
[ 13.  83.]
[ 61.  66.]
[ 93.  17.]
[  7.  64.]
[ 33.  66.]
[ 68.  55.]
[ 31. 100.]
[ 69.  71.]
[ 59.  75.]
[ 18.  41.]
[ 95.  78.]
[ 45.   3.]
[ 37.  26.]
[ 93.   1.]
[ 68.  97.]
[ 31. 100.]
```

Gambar 3.1.3.3 Hasil Pengujian 128 Titik pada Bidang 2 Dimensi

```
 [ 82.  45.]
 [ 28.  30.]
 [ 30.  93.]
 [ 77.  81.]
 [ 11.  22.]
 [  9.  48.]
 [ 66.  22.]
 [  3.  35.]
 [ 41.  10.]
 [ 63.  67.]
 [ 29.  76.]
 [ 58.  85.]
 [ 60.  82.]
 [ 51.  86.]
 [ 44.  97.]
 [ 26. 100.]]
------------------------------------
Divide and Conquer:
The closest distance is:  (0.0, 3028)
The closest points are:  [ 31. 100.]  and  [ 31. 100.]
The number of Euclidean calls:  13912
The time taken:  190.09 ms
------------------------------------
Brute Force:
The closest distance is:  (0.0, 21724)
The closest points are:  [ 31. 100.]  and  [ 31. 100.]
The number of Euclidean calls:  22040
The time taken:  37.18 ms
------------------------------------
```

Gambar 3.1.3.4 Hasil Pengujian 128 Titik pada Bidang 2 Dimensi

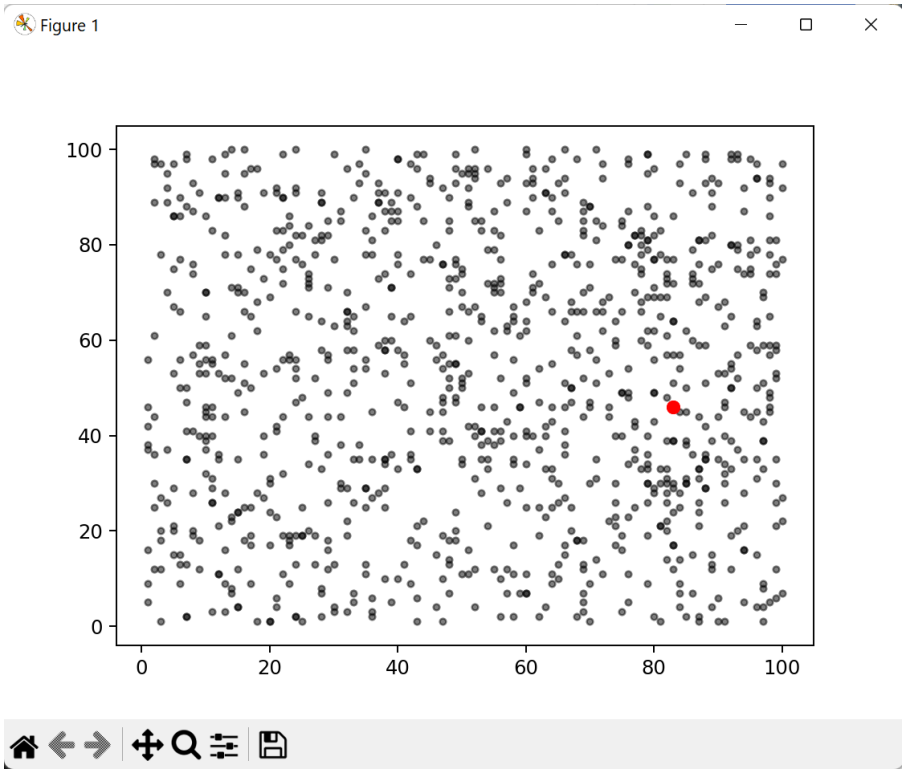


Gambar 3.1.3.5 Visualisasi Hasil Pengujian 128 Titik pada Bidang 2 Dimensi

3.1.4 Pengujian 1000 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
2
Enter the number of points:
1000
[[91. 40.]
 [71. 31.]
 [23. 86.]
 ...
 [89. 88.]
 [43. 33.]
 [ 3. 27.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (0.0, 8395)
The closest points are:  [59. 46.]  and  [59. 46.]
The number of Euclidean calls:  991671
The time taken:  12887.66 ms
-----------------------------------
Brute Force:
The closest distance is:  (0.0, 1004377)
The closest points are:  [83. 46.]  and  [83. 46.]
The number of Euclidean calls:  1491171
The time taken:  2260.2 ms
-----------------------------------
```

Gambar 3.1.4.1 Hasil Pengujian 1000 Titik pada Bidang 2 Dimensi



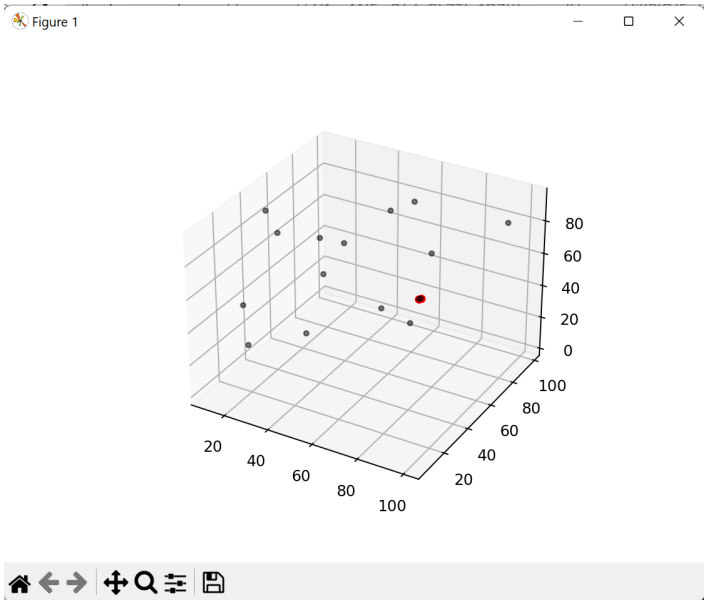
Gambar 3.1.4.2 Visualisasi Hasil Pengujian 1000 Titik pada Bidang 2 Dimensi

3.2. Hasil Pengujian 3 Dimensi

3.2.1 Pengujian 16 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
3
Enter the number of points:
16
[[ 30.  26.  93.]
 [ 27.  63.  66.]
 [ 11.  48.  86.]
 [ 82.  46.  61.]
 [ 42.  40.  64.]
 [ 56.  89.  83.]
 [ 44.  23.  40.]
 [ 21.  14.  54.]
 [100.  84.  89.]
 [ 36.  98.   2.]
 [ 28.   6.  38.]
 [ 78.  52.  55.]
 [ 44.  91.  72.]
 [ 88.  26.  62.]
 [ 52.  39.  87.]
 [ 61.  95.  49.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (9.38083151964686, 32)
The closest points are:  [82. 46. 61.]  and  [78. 52. 55.]
The number of Euclidean calls:  34
The time taken:  2.0 ms
-----------------------------------
Brute Force:
The closest distance is:  (9.38083151964686, 84)
The closest points are:  [82. 46. 61.]  and  [78. 52. 55.]
The number of Euclidean calls:  154
The time taken:  1.01 ms
-----------------------------------
```

Gambar 3.2.1.1 Hasil Pengujian 16 Titik pada Bidang 3 Dimensi



Gambar 3.2.1.2 Visualisasi Hasil Pengujian 16 Titik pada Bidang 3 Dimensi

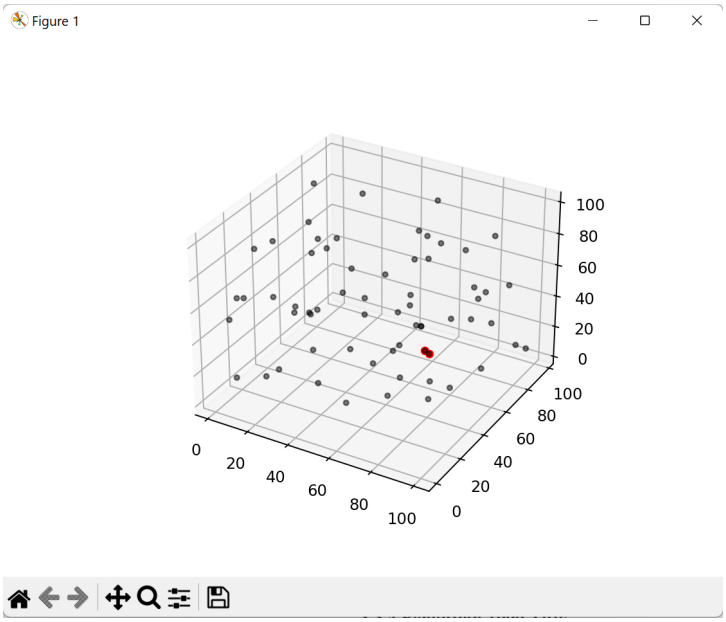
3.2.2 Pengujian 64 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
3
Enter the number of points:
64
[[ 30.  71.  99.]
 [ 49.   4.  31.]
 [ 16.  52.   5.]
 [ 32.   1.  35.]
 [ 20.  89.  16.]
 [ 95.  91.  12.]
 [ 82.  94.  76.]
 [ 54.  60.  39.]
 [ 18.  49.  71.]
 [ 39.  47.  70.]
 [ 68.  32.  39.]
 [ 63.  70.  70.]
 [  1.  15.   7.]
 [  6.  80.  50.]
 [ 47.  51.   9.]
 [ 94.  17.  41.]
 [ 78.  20.  55.]
 [  9.  61.  80.]
 [ 32.  69.  23.]
 [ 59.  13.  50.]
 [  8.  12.  63.]
 [ 52.  81.  77.]
 [ 42.  95.  46.]
 [ 53.  72.  35.]
 [ 73.  97.  30.]
 [ 88.  45.  14.]
 [ 43.  68.  54.]
 [ 49.  89.   8.]
 [  1.  31.  79.]
 [  8.  27.   2.]
 [ 61.  68.  27.]
 [ 68.  59.  20.]
```

Gambar 3.2.2.1 Hasil Pengujian 64 Titik pada Bidang 3 Dimensi

```
 [  6.   4.  54.]
 [ 56.  90.  92.]
 [ 50.  92.  66.]
 [ 11.  85.  19.]
 [ 69.  26.  15.]
 [ 67.  48.  63.]
 [ 98.  56.  71.]
 [ 63.  60.  34.]
 [ 77.  77.  76.]
 [ 27.  21.  55.]
 [ 88.  45.  58.]
 [ 24.  37.  43.]
 [  1.  45.  75.]
 [ 50.  24.   4.]
 [ 35.  21.  57.]
 [  3.  75.  94.]
 [ 69.  61.  17.]
 [  4.  76.  58.]
 [ 19.  34.  47.]
 [ 61.  50.   6.]
 [ 86.  85.  27.]
 [ 75.  87.  24.]
 [ 90.  69.  10.]
 [ 16.  22.  60.]
 [ 93.  87.  53.]
 [ 75.  59.  91.]
 [ 98.  95.   8.]
 [  7.  70.  16.]
 [  1.  17.  57.]
 [  5.  90.  50.]
 [ 85.  32.  15.]
 [100.  42.  84.]]
------------------------------------
Divide and Conquer:
The closest distance is:  (3.7416573867739413, 2983)
The closest points are:  [68. 59. 20.]  and  [69. 61. 17.]
The number of Euclidean calls:  3494
The time taken:  60.24 ms
------------------------------------
Brute Force:
The closest distance is:  (3.7416573867739413, 4999)
The closest points are:  [68. 59. 20.]  and  [69. 61. 17.]
The number of Euclidean calls:  5510
The time taken:  11.33 ms
------------------------------------
```

Gambar 3.2.2.2 Hasil Pengujian 64 Titik pada Bidang 3 Dimensi



Gambar 3.2.2.3 Visualisasi Hasil Pengujian 64 Titik pada Bidang 3 Dimensi

3.2.3 Pengujian 128 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
3
Enter the number of points:
128
[[ 12.  97.  79.]
 [ 69.  21.  44.]
 [ 14.  83.  94.]
 [ 28.   2.   7.]
 [ 77.   6.  77.]
 [ 45.   1.  16.]
 [ 97.  51.  53.]
 [ 25.  27.  17.]
 [ 13.  29.  48.]
 [ 27.  49.  71.]
 [ 59.  95.  54.]
 [ 70.   5.  12.]
 [ 41.  24.  37.]
 [ 47.  32.  14.]
 [ 79.   2.  27.]
 [  1.  56.  78.]
 [ 52.  24.   1.]
 [ 10.   1.  68.]
 [ 49.  21.  24.]
 [ 53.  24.   3.]
 [ 51.  15.  92.]
 [100.  14.  99.]
 [ 47.  66.  12.]
 [ 95.  81.  83.]
 [ 27.  78.  88.]
 [ 92.  10.  91.]
 [ 12.  48.  94.]
 [ 93.  17.  78.]
 [ 23.   7.  86.]
 [ 99.  91.  29.]
 [ 42.  11.  64.]
```

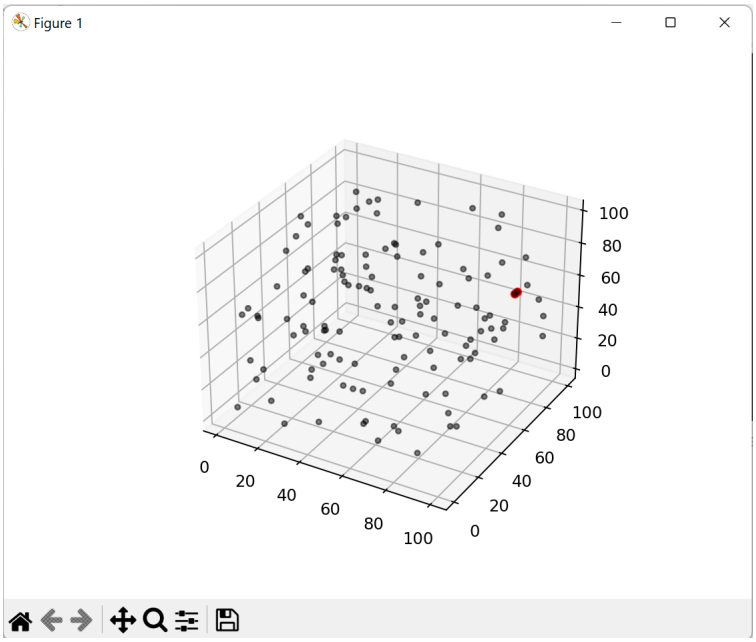
Gambar 3.2.3.1 Hasil Pengujian 128 Titik pada Bidang 3 Dimensi

```
[ 51.  52.  93.]
[  4.  24.  16.]
[ 17.   2.  68.]
[ 26.  71.  45.]
[ 20.  63.  52.]
[ 45.  81.  39.]
[ 24.  52.  29.]
[ 65.  56.  30.]
[ 34.  38.   8.]
[ 71.  76.  22.]
[ 21.  89.  88.]
[ 20.  31.  64.]
[ 52.  42.  97.]
[ 86.  56.  65.]
[ 76. 100.  19.]
[ 19.  75.  90.]
[ 40.  67.  31.]
[ 69.  56.  73.]
[ 68. 100.  50.]
[ 78.  10.  84.]
[ 78.  24.  84.]
[ 92.  53.  14.]
[ 22.  54.  99.]
[ 97.  15.  25.]
[ 63.  53.  92.]
[ 59.   6.  54.]
[ 10.  12.  60.]
[ 89.   3.  14.]
[ 56.  64.  68.]
[ 35.  81.  62.]
[ 12.  74.  59.]
[ 39.  43.  69.]
[ 10.  78.  43.]
[ 15.  65.  66.]
[ 99.  56.  61.]
[ 86.  21.  52.]
[ 77.   9.  18.]
[ 74.  28.  62.]
[ 34. 100.  84.]
[100.  84.  57.]
[ 35.  21.  34.]
[  1.  96.  66.]
[100.  73.  73.]
[ 76.  93.  87.]
[ 79.   6.  78.]
[ 98.  93.  40.]
[  4.  82.  48.]
```

Gambar 3.2.3.2 Hasil Pengujian 128 Titik pada Bidang 3 Dimensi

```
 [ 24.  74.  56.]
 [  2.   7.   3.]
 [ 68.  83.  59.]
 [ 25.  77.  48.]
 [ 99.  64.  73.]
 [ 39.  94.  18.]
 [ 66.  81.   8.]
 [ 86.  31.  29.]
 [ 62.  29.  27.]
 [ 96.  50.  60.]
 [  5.  42.  79.]
 [ 95.  61.  13.]
 [ 28.  22.  89.]
 [ 90.  48.  44.]
 [ 31.  86.  25.]
 [  1.  60.  88.]
 [  1.  89.  66.]
 [ 79.  90.  98.]
 [ 82.  41.   8.]
 [  3.  57.  21.]
 [ 16.  11.  10.]
 [ 63.  56.  60.]
 [  1.  19.  24.]
 [ 70.  82.   9.]
 [  6.  70.  11.]
 [  3.  20.  12.]
 [ 52.  49.  96.]
 [ 51.  32.  78.]
 [ 65.  62.  92.]
 [ 88.  37.   5.]
 [ 93.  50.  65.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (2.0, 4419)
The closest points are:  [99. 66. 73.]  and  [99. 64. 73.]
The number of Euclidean calls:  13134
The time taken:  189.54 ms
-----------------------------------
Brute Force:
The closest distance is:  (2.0, 20675)
The closest points are:  [99. 66. 73.]  and  [99. 64. 73.]
The number of Euclidean calls:  21262
The time taken:  42.71 ms
-----------------------------------
```

Gambar 3.2.3.3 Hasil Pengujian 128 Titik pada Bidang 3 Dimensi

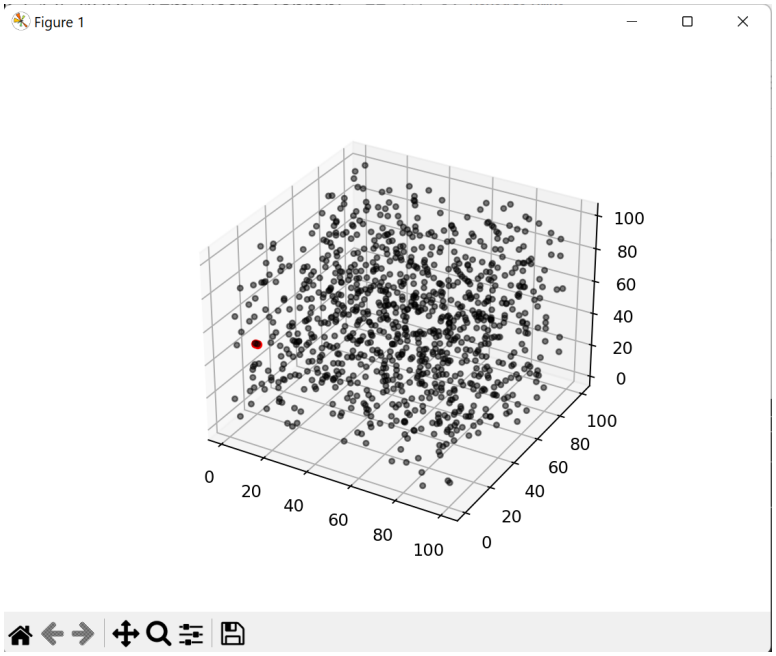


Gambar 3.2.3.4 Visualisasi Hasil Pengujian 128 Titik pada Bidang 3 Dimensi

3.2.4 Pengujian 1000 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
3
Enter the number of points:
1000
[[70. 29. 81.]
 [64. 30. 54.]
 [19. 67. 43.]
 ...
 [63. 32. 17.]
 [28. 19. 91.]
 [ 5. 84. 38.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (1.0, 762434)
The closest points are:  [ 5. 13. 45.]  and  [ 6. 13. 45.]
The number of Euclidean calls:  993475
The time taken:  14103.76 ms
-----------------------------------
Brute Force:
The closest distance is:  (1.0, 1261934)
The closest points are:  [ 5. 13. 45.]  and  [ 6. 13. 45.]
The number of Euclidean calls:  1492975
The time taken:  2717.63 ms
-----------------------------------
```

Gambar 3.2.4.1 Hasil Pengujian 1000 Titik pada Bidang 3 Dimensi



Gambar 3.2.4.2 Visualisasi Hasil Pengujian 1000 Titik pada Bidang 3 Dimensi

3.3 Hasil Pengujian N-Dimensi

3.3.1 Pengujian 16 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
5
Enter the number of points:
16
[[92. 45. 88. 69. 65.]
 [25. 69. 42. 44. 90.]
 [41. 13. 14. 91. 43.]
 [16. 75. 18. 10. 75.]
 [32. 74. 49. 24. 50.]
 [23. 61. 62. 55. 27.]
 [ 5. 83. 55. 65. 29.]
 [ 9. 37. 18. 31. 81.]
 [49. 74. 25. 83. 70.]
 [11. 93. 82. 66. 42.]
 [70. 22. 87. 17. 82.]
 [57. 30. 92. 14. 91.]
 [19. 33. 78. 17.  8.]
 [ 7.  6. 29.  5. 58.]
 [66.  4. 21. 12. 65.]
 [73. 84.  2. 32. 45.]]
-----------------------------------
Divide and Conquer:
The closest distance is:  (18.65475810617763, 28)
The closest points are:  [70. 22. 87. 17. 82.]  and  [57. 30. 92. 14. 91.]
The number of Euclidean calls:  52
The time taken:  2.0 ms
-----------------------------------
Brute Force:
The closest distance is:  (18.65475810617763, 158)
The closest points are:  [70. 22. 87. 17. 82.]  and  [57. 30. 92. 14. 91.]
The number of Euclidean calls:  172
The time taken:  0.82 ms
-----------------------------------
Cannot plot the points in  5  dimensions
```

Gambar 3.3.1.1 Hasil Pengujian 16 Titik pada Bidang N Dimensi

3.3.2 Pengujian 64 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
10
Enter the number of points:
64
[[ 22.  71.  84.  28.  71.  86.  80.  18.   1.  64.]
 [ 39.  71.  80.  36.  47.  51.  26.  37.  77.  23.]
 [ 65.  92.   1.  70.  44.  61.  99.  15.  95.  45.]
 [ 72.  12.  96.  24.  43.   4.  36.  79.  43.  95.]
 [  1.  84.  95.  84.   8.  46.  87.  68.  10.  49.]
 [ 18.  36.  14.  82.  15.  75.  46.  24.  42.  59.]
 [ 74.  42.   2.  10.  47.  30.  10.  51.  34.  43.]
 [ 32.  28.  78.  80.  34.  14.  54.  24.  47.  83.]
 [ 25.  30.  70.  30.   7.  96.  55.  30.  11.   8.]
 [ 23.  69.  72.  32.  60.  73.  45.  10.  38.  35.]
 [ 32.  72.   5.  54.  69.  65.  14.  19.  59. 100.]
 [ 34.  81.  34.   9.  17.  94.  52.  82.  78.  39.]
 [ 33.  97.  94.  84.  79.  24.  46.  84.  92.  27.]
 [ 61.  38.  17.  81.  85.  14.  68.  84.  30.  31.]
 [ 86.  11.  12.  75.  82.  28.  21.  64.   9.  45.]
 [ 87.  49.  96.  61.  12.  87.  77.  63.  86.  39.]
 [ 94.  84.  27.  74.  93.  95.  49.  89.  73.  21.]
 [ 91.  46.  32.  26.  85.  19.  48.  81.  77.  66.]
 [ 86.  57.  36.  65.  67.  37.  19.  55.  34. 100.]
 [  5.  62.  55.  50.   8.   6.  89.  27.   8.  22.]
 [ 95.  86.  70.  84.  73.  72.  38.   1.  46.  30.]
 [ 17.  67.  84.  62.  33.  57.  46.   3.  40.  20.]
 [ 73.  96.  17.  87.  62.  91.  41.  79.  80.   2.]
 [ 55.  15.  34.   1.  48.  43.  79.  92.  74.  26.]
 [ 34.  72.  94.  77.  58.  78.  24.  88.  82.  92.]
 [  7.  17.  80.  23.  99.  82.  29.  63.  24.   6.]
 [ 66.  99.  66.   1.  10.  81.  24.  21.  99.  23.]
 [ 78.   8.  98.  58.  36.  10.   8.  55.  41.  93.]
 [ 55.  43.  69.  17.  12. 100.   7. 100.  24.  62.]
 [ 29.  18.  92.  68.  51.  71.  12.  20.  20.  74.]
```

Gambar 3.3.2.1 Hasil Pengujian 64 Titik pada Bidang N Dimensi

```
 [ 76.  89.  22. 100.  19.  42.  18.   8.  81.  48.]
 [ 60.  43.  29.  31.  37.  68.  96.  48.  22.  45.]
 [ 45.  99.  37.  13.  41.  45.  83.  57.  28.  54.]
 [ 73.  38.  70.  21.   8.  38.  48. 100.  98.  21.]
 [  6.  54.  98.  88.  23.   1.  96.  33.  79.  26.]
 [ 66.  18.  66.  34.  86.  11.  72.  57.  13.   4.]
 [ 68.   2.  12.  84.  73.  18.  72.   5.  60.  76.]
 [ 16.  76.  30.  17.  63.  38.  16.  77.  64.  16.]
 [  9.  55.  44.  10.  27.  84.  38.  71.  76.  34.]
 [ 10.  64.  28.   3.  10.  56.  26.  21.  68.  57.]
 [ 33.  36.   4.  89.  37.  45.  23.  68.  86.  27.]
 [ 95.  54.   6.   3.  11.  47.  36.  86.  81.  73.]
 [ 56.  67.  25.  42.  17.  25.  60.  62.  55.  91.]
 [ 69.  98.  64.  44.  66.  64.  43.  22.  40.  36.]
 [ 86.  93.  17.  30.  37.  66. 100.  97.  26.  65.]
 [  6.  99.  57.  65.  76.  29.  21.  38.  59.  28.]
 [ 91.  93.  90.  22.  67.  34.  13.  99.  34.  68.]
 [ 13.   6.  67.  56.  56.   9.  55.  19.  39.  33.]
 [ 73.  64.   2.  85.  35.  59.  14.  62.  93.  23.]
 [ 90.  27.  90.  81.  37.  87.  60.  20.  15.  98.]
 [ 24.   4.  89.  19.  62.  28.  54.   3.  84.  53.]
 [ 34.  76.  88.  14.  76.  24.  81.  47.  25.  44.]
 [  5.  95.   9.  98.  62.  20.  81.  44.  19.  33.]
 [ 66.  94.  71.  36.  85.   9.   6.  94.  61.  14.]
 [ 42.  29.  39.  77.  25.  72.  96.  51.  13.  49.]
 [ 16.  95.   4.  48.  46.  79.  85.  29.  54.  37.]
 [ 72.   3.  23.  68.  47.  84.  63.  40.   1.  70.]
 [ 34.  84.  55.  37.  81.  35.  23.  95.  62.  95.]
 [  4.  97.  74.  34.  91.  84.  99.  63.  92.   9.]
 [ 11.  55.  44.  11.  91.  78.  72.  82.  47.  78.]
 [ 64.  38.  95.   6.  11.   2.  83.  37.  10.  35.]
 [ 83.  30.   7.  27.  80.  22.  67.  12.  95.  41.]
 [ 75.  94.  11.  67.   2.   5.  46.  39.  19.  74.]
 [ 88. 100.  12.  84.  49.  32. 100.  29.  87.  17.]]
-----------------------------------
```

Gambar 3.3.2.2 Hasil Pengujian 64 Titik pada Bidang N Dimensi

```
Divide and Conquer:
The closest distance is:  (44.13615298142782, 2487)
The closest points are:  [34. 81. 34.  9. 17. 94. 52. 82. 78. 39.]  and  [ 9. 55. 44. 10. 27. 84. 38. 71. 76. 34.]
The number of Euclidean calls:  3838
The time taken:  87.74 ms
-----------------------------------
Brute Force:
The closest distance is:  (44.13615298142782, 4503)
The closest points are:  [34. 81. 34.  9. 17. 94. 52. 82. 78. 39.]  and  [ 9. 55. 44. 10. 27. 84. 38. 71. 76. 34.]
The number of Euclidean calls:  5854
The time taken:  22.59 ms
-----------------------------------
Cannot plot the points in  10  dimensions
```

Gambar 3.3.2.3 Hasil Pengujian 64 Titik pada Bidang N Dimensi

3.3.3 Pengujian 128 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
15
Enter the number of points:
128
[[12. 23. 56. ... 89. 60. 72.]
 [52. 10. 27. ... 47. 63. 97.]
 [57. 88. 80. ... 47. 72. 23.]
 ...
 [33. 79. 50. ... 50. 70. 51.]
 [17. 78. 90. ... 42. 35. 95.]
 [37. 45. 72. ... 69.  8. 90.]]
------------------------------------
Divide and Conquer:
The closest distance is:  (68.35202996254024, 672)
The closest points are:  [27. 88. 79. 64. 85. 36. 89. 20. 81. 51. 22. 66.  3. 98. 26.]  and  [ 6. 55. 53. 68. 73. 29. 80. 19. 77. 34. 15. 91. 29
. 92.  4.]
The number of Euclidean calls:  15808
The time taken:  372.25 ms
------------------------------------
Brute Force:
The closest distance is:  (68.35202996254024, 16928)
The closest points are:  [27. 88. 79. 64. 85. 36. 89. 20. 81. 51. 22. 66.  3. 98. 26.]  and  [ 6. 55. 53. 68. 73. 29. 80. 19. 77. 34. 15. 91. 29
. 92.  4.]
The number of Euclidean calls:  23936
The time taken:  134.79 ms
------------------------------------
Cannot plot the points in  15  dimensions
```

Gambar 3.3.3.1 Hasil Pengujian 128 Titik pada Bidang N Dimensi

3.3.4 Pengujian 1000 Titik

```
START? (Y/N)
y
Enter the number of dimensions:
100
Enter the number of points:
1000
[[15. 59. 47. ... 15. 78. 56.]
 [96. 29. 13. ... 19. 18. 82.]
 [97. 42. 92. ... 21. 17. 95.]
 ...
 [29. 63. 14. ... 80. 33. 46.]
 [11. 79. 38. ... 48. 58. 43.]
 [27. 41. 60. ... 65. 28. 11.]]
------------------------------------
Divide and Conquer:
The closest distance is:  (295.37772427859215, 633646)
The closest points are:  [82. 76. 49.  2. 56. 89. 57. 57. 49. 11. 80. 70. 92. 54. 62. 72. 15. 80.
 84. 26. 92. 29. 71. 49. 88.  8. 12. 51. 85. 81. 21.  7. 18. 16. 81. 31.
 83. 27. 46. 33. 86. 23. 53. 88. 23. 15. 26. 31. 51.  6. 77. 78. 30. 48.
 94. 20. 10. 77. 17. 45.  5. 49. 59. 89. 46. 88. 22. 96. 14. 57. 67. 14.
 94. 47. 10. 50. 19. 40. 27. 55. 53. 86. 32. 63. 15. 46. 30. 14. 47. 57.
 36. 97. 79. 45. 77. 17. 15. 13. 81. 19.]  and  [ 25.  37.  83.  30.  22.  51.  92.  38.  74.  29.  84.  68. 100.  92.
  36.  55.  75.  48.  73.  25.  37.  68.  95.  58.  60.  16.  21.  49.
  57.  72.  11.   4.  76.  56.  92.  24.  28.  12.  36.  39.  91.  19.
  25.  96.  34.  72.   1.  27.  83.   3.  40.  65.   7.  42.  50.  67.
  33.  81.  31.  58.  19.  38.  87.  46.  62.  60.  40.  13.  18.  22.
  69.   4.  74.  35.  79.  83.  58.  85.  43.  30.   4.  92.  27.  55.
   2.  56.  27.  62.  73.  39.  66.  37.  84.  15.  39.  32.  25.  61.
  99.   7.]
The number of Euclidean calls:  994044
The time taken:  141448.34 ms
------------------------------------
```

Gambar 3.3.4.1 Hasil Pengujian 1000 Titik pada Bidang N Dimensi

```
-----------------------------------
Brute Force:
The closest distance is:  (295.37772427859215, 1133146)
The closest points are:  [82. 76. 49.  2. 56. 89. 57. 57. 49. 11. 80. 70. 92. 54. 62. 72. 15. 80.
 84. 26. 92. 29. 71. 49. 88.  8. 12. 51. 85. 81. 21.  7. 18. 16. 81. 31.
 83. 27. 46. 33. 86. 23. 53. 88. 23. 15. 26. 31. 51.  6. 77. 78. 30. 48.
 94. 20. 10. 77. 17. 45.  5. 49. 59. 89. 46. 88. 22. 96. 14. 57. 67. 14.
 94. 47. 10. 50. 19. 40. 27. 55. 53. 86. 32. 63. 15. 46. 30. 14. 47. 57.
 36. 97. 79. 45. 77. 17. 15. 13. 81. 19.]  and  [ 25.  37.  83.  30.  22.  51.  92.  38.  74.  29.  84.  68. 100.  92.
  36.  55.  75.  48.  73.  25.  37.  68.  95.  58.  60.  16.  21.  49.
  57.  72.  11.   4.  76.  56.  92.  24.  28.  12.  36.  39.  91.  19.
  25.  96.  34.  72.   1.  27.  83.   3.  40.  65.   7.  42.  50.  67.
  33.  81.  31.  58.  19.  38.  87.  46.  62.  60.  40.  13.  18.  22.
  69.   4.  74.  35.  79.  83.  58.  85.  43.  30.   4.  92.  27.  55.
   2.  56.  27.  62.  73.  39.  66.  37.  84.  15.  39.  32.  25.  61.
  99.   7.]
The number of Euclidean calls:  1493544
The time taken:  58756.59 ms
-----------------------------------
```

Gambar 3.3.4.2 Hasil Pengujian 1000 Titik pada Bidang N Dimensi

BAB IV
LAMPIRAN

4.1 Tautan Repository

https://github.com/goodgirlwannabe/Tucil2_13521006.git

4.2 Tabel Penilaian

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa ada kesalahan	✔	
2. Program berhasil *running*	✔	
3. Program dapat menerima masukan dan menuliskan luaran	✔	
4. Luaran program sudah benar (solusi closest pair benar)	✔	
5. Bonus 1 dikerjakan	✔	
6. Bonus 2 dikerjakan	✔	