

Penerapan Algoritma Greedy untuk Membentuk Kelompok Belajar Berdasarkan Kompetensi Mahasiswa

Kandida Edgina Gunawan - 13521155
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13521155@std.stei.itb.ac.id

Abstract— Algoritma *Greedy* adalah algoritma yang memecahkan persoalan langkah per langkah sedemikian sehingga pada setiap langkah akan diambil pilihan terbaik yang dapat diperoleh saat itu dengan harapan dengan memilih optimum lokal pada setiap langkah akan berakhir dengan optimum global. Algoritma *Greedy* dapat digunakan untuk menyelesaikan beberapa persoalan, yaitu persoalan penukaran uang, penjadwalan aktivitas, dan sebagainya. Pada makalah ini akan dijelaskan mengenai penerapan algoritma *greedy* untuk menentukan kombinasi orang-orang terbaik yang cocok untuk dijadikan teman kelompok belajar bagi seseorang apabila diketahui nilai semua mata kuliah dan daftar orang yang tidak disukai oleh setiap orang.

Kata Kunci—*kelompok, greedy, nilai mata kuliah, daftar orang, kompetensi*

I. PENDAHULUAN

Algoritma *Greedy* adalah algoritma yang paling populer dan cukup sederhana untuk memecahkan persoalan optimasi. Algoritma *Greedy* merupakan algoritma yang memecahkan persoalan langkah per langkah sehingga pada setiap langkah akan diambil pilihan terbaik pada saat itu, tanpa memperhatikan konsekuensi yang akan datang ke depannya sembari berharap bahwa dengan selalu memilih optimum lokal pada tiap langkah, kita akan berakhir dengan optimum global untuk menyelesaikan persoalan tersebut. Algoritma *Greedy* telah terbukti dapat menyelesaikan berbagai masalah, yaitu persoalan penukaran uang, persoalan pemilihan aktivitas, persoalan minimisasi waktu dalam sistem, dan masih banyak lagi.

Persoalan pembentukan kelompok belajar merupakan persoalan yang sering terjadi di kalangan mahasiswa terutama di kalangan mahasiswa Informatika yang sering diharuskan untuk membentuk kelompok untuk tugas besar misalnya. Tentunya dalam pembentukan suatu kelompok, kita menginginkan agar setiap anggota yang ada di dalamnya dapat memberikan kontribusi yang besar bagi kelompok. Untuk itu, setiap anggota harus memiliki kapabilitas dan pengetahuan yang baik mengenai subjek yang ingin dipecahkan dalam kelompok. Namun, kemampuan akademik yang baik dari tiap anggotanya saja dianggap tidak cukup untuk dapat

menciptakan kelompok belajar yang baik. Suatu kelompok belajar yang baik dapat tercipta juga hanya apabila setiap anggotanya bisa nyaman bekerja bersama. Dengan adanya rasa nyaman dalam bekerja bersama, produktivitas kelompok pasti juga akan meningkat. Hal ini juga akan mengurangi kemungkinan adanya cekcok selama proses kerja tim.

Untuk memecahkan persoalan pembentukan kelompok belajar tersebut, penulis ingin menerapkan algoritma *greedy*. Penulis berharap dengan memilih anggota satu per satu dengan kualifikasi terbaik, nantinya akan terbentuk suatu kombinasi anggota tim yang tak hanya kompeten, namun juga dapat bekerja sama dengan baik. Implementasi dari persoalan pembentukan kelompok belajar ini akan dibuat dalam bahasa pemrograman *python*.

II. DASAR TEORI

A. Algoritma Greedy

Algoritma *Greedy* adalah metode yang paling populer dan sederhana untuk memecahkan persoalan optimasi. Terdapat dua macam persoalan optimasi, yaitu maksimasi dan minimasi. Algoritma *Degree* memecahkan suatu persoalan secara langkah per langkah dengan setiap langkahnya akan mengambil pilihan terbaik yang dapat diperoleh tanpa memperhatikan konsekuensi ke depannya sembari berharap bahwa dengan selalu memilih optimum lokal pada tiap langkah, kita akan berakhir dengan optimum global.

Terdapat beberapa elemen dalam algoritma *greedy*, yaitu sebagai berikut:

1. Himpunan kandidat, C : berisi kandidat yang akan dipilih pada setiap langkah
2. Himpunan solusi, S : berisi kandidat yang sudah dipilih
3. Fungsi solusi : menentukan apakah himpunan kandidat yang dipilih sudah memberikan solusi
4. Fungsi seleksi: memilih kandidat berdasarkan strategi tertentu. Strategi *greedy* ini bersifat heuristik.

5. Fungsi kelayakan: memeriksa apakah kandidat yang dipilih dapat dimasukkan ke dalam himpunan solusi (layak atau tidak)
6. Fungsi objektif: memaksimumkan atau meminimumkan

Skema umum dari algoritma *greedy* dapat dilihat pada *pseudocode* berikut.

```

function greedy(C : himpunan_kandidat) → himpunan_solusi
  / Mengembalikan solusi dari persoalan optimasi dengan algoritma greedy /
  Deklarasi
  x : kandidat
  S : himpunan_solusi

  Algoritma:
  S ← {} / inisialisasi S dengan kosong /
  while (not SOLUSI(S)) and (C ≠ {} ) do
    x ← SELEKSI(C) / pilih sebuah kandidat dari C /
    C ← C - {x} / buang x dari C karena sudah dipilih /
    if LAYAK(S ∪ {x}) then / x memenuhi kelayakan untuk dimasukkan ke dalam himpunan solusi /
      S ← S ∪ {x} / masukkan x ke dalam himpunan solusi /
    endif
  endwhile
  / SOLUSI(S) or C = {} /

  if SOLUSI(S) then / solusi sudah lengkap /
    return S
  else
    write("tidak ada solusi")
  endif

```

Contoh masalah yang dapat diselesaikan dengan menggunakan algoritma *greedy*, yaitu persoalan penukaran uang, persoalan pemilihan aktivitas, minimisasi waktu di dalam sistem, persoalan *knapsack*, dan penjadwalan job dengan tenggat waktu.

B. Kompetensi Mahasiswa

Kompetensi merupakan dari pengetahuan, keterampilan, dan atribut kepribadian seseorang yang digunakan oleh seseorang untuk meningkatkan kinerjanya dan memberikan kontribusi bagi keberhasilan organisasinya. Kompetensi juga dapat diartikan sebagai kapasitas yang terdapat pada diri seseorang yang membuat dirinya mampu untuk memenuhi apa yang dibutuhkan atau diminta oleh suatu organisasi, ia mampu mencapai hasil yang diharapkan oleh organisasi tersebut. Beberapa indikator dari kompetensi, yakni sebagai berikut:

1. Pengetahuan
Pengetahuan yang dimaksud di sini dapat berupa kemampuan kognitif seseorang
2. Pemahaman
Pemahaman yang dimaksud di sini adalah kedalaman konsep yang diyakini oleh individu tersebut
3. Kemampuan
Kemampuan yang dimaksud di sini adalah sesuatu yang dimiliki oleh individu tersebut untuk dapat menyelesaikan tugas-tugas yang diberikan kepadanya dengan baik.
4. Sikap
Sikap adalah kecenderungan dari suatu individu untuk merespons suatu kejadian atau memperlihatkan suatu perilaku tertentu
5. Minat
Minat adalah ketertarikan terhadap bidang atau topik tertentu.

Dalam lingkup kompetensi mahasiswa, terdapat beberapa cara yang dapat dilakukan untuk mengukur kompetensi dari mahasiswa, yaitu sebagai berikut:

1. Ujian tertulis
Ujian tertulis merupakan metode yang paling umum dilakukan untuk menilai kompetensi dari mahasiswa. Ujian ini dapat memiliki beberapa format, seperti pilihan ganda, essay, ataupun studi kasus.
2. Praktikum atau simulasi
Penilaian praktikum atau simulasi ini mewajibkan mahasiswa untuk secara langsung mencoba untuk menerapkan teori yang telah mereka pelajari selama ini.
3. Pemberian proyek
Mahasiswa diberikan tugas atau proyek yang secara nyata mencerminkan situasi nyata pada bidang yang relevan dengan jurusan mereka. Tugas yang diberikan biasanya akan menguji kemampuan dan keterampilan yang mereka miliki. Dalam jurusan Informatika, tugas proyek seperti ini dapat kita jumpai pada tugas besar 3 strategi algoritma misalnya yang mengharuskan mahasiswa untuk membuat suatu *website*. Tugas ini memberikan cerminan pekerjaan dari seorang *web developer*.
4. Penilaian *peer-to-peer*
Penilaian *peer-to-peer* adalah penilaian yang diberikan oleh seorang mahasiswa terhadap mahasiswa yang lainnya. Biasanya penilaian *peer-to-peer* ini dilakukan setelah berakhirnya proyek tugas besar dari mahasiswa. Tujuan dari penilaian ini adalah tiap mahasiswa dapat memberikan penilaian terhadap pekerjaan teman sekelompoknya.

C. Persoalan Pembentukan Kelompok Belajar

Kelompok adalah sekumpulan orang yang saling berinteraksi satu sama lain untuk mencapai suatu tujuan yang sama. Syarat terbentuknya suatu kelompok adalah adanya kesadaran sebagian dari kelompok yang bersangkutan, adanya kesamaan antarindividu, adanya hubungan timbal balik yang saling menguntungkan antaranggota, serta adanya faktor pengikat.

Secara umum, kelompok dapat dibagi menjadi 4 jenis, yaitu:

1. Kelompok primer
Kelompok primer adalah kelompok yang seluruh anggotanya sering bertemu secara langsung dan sudah mengenal secara dekat. Contoh kelompok primer adalah keluarga, kelompok belajar, kelompok agama, dan kelompok bermain.
2. Kelompok sekunder
Kelompok sekunder adalah kelompok yang berhubungan secara tidak langsung dan kurang bersifat kekeluargaan, sertanya hubungannya cenderung bersifat objektif. Contoh kelompok sekunder adalah partai politik.
3. Kelompok formal
Kelompok formal adalah kelompok yang didukung oleh peraturan anggaran dasar dan anggaran rumah tangga.

4. Kelompok informal

Kelompok informal adalah kelompok yang terbentuk berdasarkan rasa suka dan tidak suka, dalam iklim psikis yang intim. Kelompok ini biasanya sering bertemu langsung dan memiliki moral yang tinggi. Status dari kelompok informal tidak resmi dan tidak didukung oleh peraturan-peraturan.

Pada makalah ini, yang akan dibahas lebih dalam adalah mengenai kelompok belajar. Kelompok belajar merupakan salah satu kebutuhan dari mahasiswa. Biasanya mahasiswa akan diminta untuk membentuk suatu kelompok belajar untuk menyelesaikan suatu tugas tertentu yang diberikan oleh dosen. Menurut Johnson dan Johnson (dalam Walgito, 2006:15), terdapat beberapa tahapan seseorang bergabung dalam suatu kelompok, yaitu:

1. *Prospective Member*, yaitu calon anggota potensial yang akan bergabung ke dalam suatu kelompok akan dievaluasi, yaitu apakah dengan bergabungnya calon anggota tersebut ke dalam kelompok yang bersangkutan akan menguntungkan calon anggota tersebut serta kelompok tersebut atau tidak
2. *New Member*, yaitu anggota baru yang masuk akan menyesuaikan diri dengan lingkungan kelompok. Anggota baru akan dibiasakan dengan status dan perannya dalam kelompok tersebut
3. *Full Member*, yaitu kelompok sudah memiliki jumlah anggota yang diinginkan sehingga tak perlu lagi diadakan rekrutmen ke dalam kelompok
4. *Marginal Member*, yaitu anggota mempunyai keraguan terhadap kelompok yang bersangkutan karena adanya cekcok dengan anggota lainnya, ataupun karena adanya norma yang tidak sesuai dengan ideologinya.
5. *Ex-member*, yaitu anggota yang sudah tidak lagi terikat pada suatu kelompok tertentu (sudah tidak lagi menjadi anggota)

Tentunya dalam proses perekrutan anggota kelompok, kita akan menyeleksi orang-orang yang kita inginkan untuk berada di dalam kelompok tersebut. Harapannya adalah kelompok yang terbentuk dapat memiliki anggota-anggota yang memiliki kompetensi yang baik. Selain daripada itu, penting bagi setiap anggota untuk dapat bekerja sama dengan nyaman antara satu dengan yang lain. Untuk itu, hal yang paling mudah untuk menilai kompetensi dari mahasiswa adalah dari nilai mahasiswa tersebut. Nilai merupakan sesuatu yang sifatnya dapat diukur sehingga kita dapat mengurutkan mahasiswa dengan kompetensi terbaik berdasarkan nilai yang dimilikinya. Perekrutan yang paling mudah adalah dengan memasukkan semua mahasiswa dengan nilai terbaik ke dalam sebuah kelompok. Kendati demikian, tak semua orang yang memiliki kompetensi tinggi akan saling cocok satu sama lain. Oleh karena itu, perlu pula dipertimbangkan relasi antara satu anggota dengan anggota lainnya dalam kehidupan nyata sebelum menjadikan mereka sebagai satu kelompok.

III. PEMBENTUKAN KELOMPOK BELAJAR MENGGUNAKAN ALGORITMA GREEDY

Masalah pembentukan kelompok belajar dengan memilih satu per satu anggota kelompok ini akan coba diselesaikan dengan menggunakan pendekatan algoritma *greedy*. Sebagai masukan nantinya akan diberikan data nilai-nilai yang dimiliki oleh setiap mahasiswa. Selain itu, pada data tiap mahasiswa, tercatat pula nama-nama orang yang tidak ia inginkan untuk sekelompok dengannya. Dari data mahasiswa tersebut, nantinya kita akan memilih nama-nama mahasiswa yang paling kompeten dan cocok untuk menjadi teman kelompok kita yang kemudian daftar nama tersebut akan menjadi keluaran yang kita harapkan.

A. Identifikasi Elemen-Elemen dalam Algoritma Greedy dalam Pemecahan Masalah Pembentukan Kelompok Belajar

1. Himpunan kandidat, C: himpunan NIM mahasiswa yang belum mendapatkan kelompok
2. Himpunan solusi: NIM-NIM mahasiswa yang terpilih menjadi anggota dari kelompok
3. Fungsi solusi: memeriksa apakah jumlah orang yang telah terpilih menjadi anggota telah sesuai dengan permintaan jumlah anggota yang diharapkan
4. Fungsi seleksi: memilih mahasiswa dengan rata-rata nilai tertinggi dari himpunan mahasiswa yang belum memiliki kelompok yang tersisa
5. Fungsi kelayakan: memeriksa apakah mahasiswa baru yang terpilih tidak disukai oleh satu atau lebih anggota yang telah terpilih sebelumnya pada kelompok atau apakah anggota baru yang terpilih ternyata tidak menyukai satu atau lebih anggota yang telah terpilih sebelumnya
6. Fungsi objektif: kompetensi kelompok yang terbentuk maksimum (nilai rata-rata tugas besar pada sebuah kelompok maksimal dengan ketentuan tak boleh ada anggota yang saling cekcok)

B. Strategi Greedy yang Diterapkan

Prekondisi :

1. Mensortir data mahasiswa yang belum mempunyai kelompok, data mahasiswa yang diperhitungkan untuk pemilihan mahasiswa ke dalam kelompok belajar hanyalah data mahasiswa yang belum mempunyai kelompok saja.
2. Menghitung nilai rata-rata yang dimiliki oleh setiap mahasiswa
3. Mengurutkan data mahasiswa berdasarkan rata-rata nilai yang dimiliki secara terurut mengecil (*descending*)

Strategi *Greedy*:

1. Memilih mahasiswa dengan nilai rata-rata tertinggi yang belum memiliki kelompok
2. Lakukan langkah 1 secara berulang sampai jumlah anggota kelompok yang diinginkan terpenuhi

- Seorang mahasiswa dinyatakan layak untuk masuk ke dalam kelompok belajar apabila anggota-anggota lain yang telah ada sebelumnya di kelompok belajar tidak ada yang membenci atau memasukkan nama mahasiswa tersebut ke dalam daftar *blacklist*. Jika mahasiswa yang berada pada urutan teratas pada daftar (mahasiswa dengan nilai rata-rata tertinggi setelah anggota-anggota kelompok sebelumnya) ternyata dibenci oleh salah satu dari anggota kelompok terpilih, nama mahasiswa pada daftar teratas tersebut tidak akan dimasukkan ke dalam kelompok dan pemeriksaan akan dilanjutkan untuk nama mahasiswa yang berada di urutan setelahnya.

C. Data yang Digunakan

Berikut ini merupakan tabel data yang akan digunakan sebagai masukan dalam implementasi solusi pembentukan kelompok belajar.

NIM	NAMA	NILAI_KUIS	NILAI_TUBES1	NILAI_TUBES2	NILAI_TUBES3	NILAI_UTS	BLACK_LIST
13521001	Ani	64	31	46	75	60	13521015;13521012
13521002	Budi	89	78	89	92	68	13521004;13521006
13521003	Siti	45	50	53	58	50	13521013
13521004	Kurnia	77	80	83	84	97	13521012
13521005	Kasih	96	88	79	67	70	13521004;13521005
13521006	Teddy	68	62	72	72	84	13521001;13521002;13521003
13521007	William	88	83	83	78	92	13521009;13521014
13521008	Matthew	76	66	89	94	69	13521006
13521009	Lisa	92	95	88	83	80	13521004;13521012;13521013
13521010	Gita	82	93	83	72	75	13521001
13521011	Feb	79	85	76	82	84	13521004
13521012	Harry	76	78	89	78	90	
13521013	Dida	100	82	93	83	94	13521009;1352101
13521014	Michael	74	79	86	93	81	
13521015	Bruno	69	66	82	82	79	13521003;13521004

Gambar 3.1. Tabel data mahasiswa

Tabel di atas memiliki beberapa atribut, yaitu NIM, NAMA, NILAI_KUIS, NILAI_TUBES1, NILAI_TUBES2, NILAI_TUBES3, NILAI_UTS, serta BLACK_LIST yang dimiliki oleh setiap mahasiswa. Setiap mahasiswa memiliki kode NIM yang unik untuk setiap mahasiswa. Nama mahasiswa yang satu bisa saja sama dengan mahasiswa yang lain sehingga yang dijadikan *key* atau pembeda dalam data ini adalah atribut NIM. Di dalam tabel di atas, dicatat pula nilai kuis, nilai tugas besar pertama, nilai tubes besar kedua, nilai tugas besar ketiga, dan nilai UTS dari tiap mahasiswa. Nilai tersebut berada di antara *range* 0-100. Selain data nilai, terdapat juga kolom BLACK_LIST yang mencatat NIM-NIM mahasiswa yang tidak disukai oleh mahasiswa tersebut. Misalnya pada *record* mahasiswa dengan NIM 13521001, kita dapat melihat bahwa terdapat 2 NIM yang berada pada kolom BLACK_LIST untuk *record* tersebut, yaitu NIM 13521015 dan 13521012. Hal ini berarti mahasiswa dengan NIM 13521001 tidak ingin sekelompok dengan mahasiswa dengan NIM 13521015 dan 13521012. Kolom BLACK_LIST untuk *record* tertentu bisa saja kosong jika mahasiswa tersebut tidak masalah sekelompok dengan siapapun.

D. Implementasi Solusi dalam Bahasa Pemrograman Python

Implementasi dari solusi dikerjakan dalam bahasa pemrograman Python pada platform google collab

A. Pemrosesan Awal Data

Pada tahap ini, data masukan akan diproses sedemikian rupa agar memenuhi prekondisi yang telah dinyatakan sebelumnya pada bagian penjelasan mengenai strategi *greedy*.

- Mengimport beberapa *library* yang digunakan seperti *pandas* dan *csv*

```
import pandas as pd
import csv
```

- Meng-*import file csv* yang digunakan sebagai data masukan, serta menambahkan kolom NILAI_RATAAN_TUBES yang menyatakan nilai rata-rata dari 3 tubes yang terdapat pada tabel

```
df = pd.read_csv("data_mahasiswa.csv")
df["NILAI_RATAAN_TUBES"] = (df["NILAI_TUBES1"] + df["NILAI_TUBES2"] + df["NILAI_TUBES3"])/3
```

- Menyortir data mahasiswa yang belum mempunyai kelompok, mengurutkan data tersebut berdasarkan NILAI_RATAAN_TUBES dengan urutan menurun (*descending*), serta memasukkan data baru yang telah disortir tersebut ke dalam variabel *toCheck*.

```
toCheck = df[df["SUBURUPA_KELOMPOK"] == False].sort_values(by="NILAI_RATAAN_TUBES", ascending=False)
```

B. Implementasi Algoritma Greedy

- Membuat fungsi *getBlackList(NIM)* sebagai fungsi bantuan yang akan mengembalikan daftar NIM mahasiswa yang tidak disukai oleh mahasiswa dengan NIM masukan

```
def getBlackList(NIM):
    result = toCheck.loc[toCheck["NIM"] == NIM, "BLACK_LIST"].str.split()
    return result.iloc[0]
```

- Membuat fungsi untuk menghitung rata-rata nilai tubes dalam satu kelompok (tak termasuk nilai dari pembuat kelompok)

```
def rata_rata_kelompok(kelompok, NIM_SAYA):
    print("hai", kelompok)
    jumlah = 0.0
    for i in kelompok:
        if i == NIM_SAYA:
            continue
        jumlah += toCheck.loc[toCheck["NIM"] == i, "NILAI_RATAAN_TUBES"].iloc[0]
    return jumlah/(len(kelompok)-1)
```

- Membuat fungsi kelayakan untuk mengecek apakah seorang mahasiswa dengan NIM

masuk ke dalam kelompok belajar. Seorang mahasiswa dinyatakan layak untuk masuk ke dalam kelompok belajar tersebut apabila mahasiswa baru yang ingin masuk tersebut tidak masuk ke dalam daftar *black list* dari mahasiswa sebelumnya yang telah masuk ke dalam kelompok belajar dan mahasiswa baru tersebut tidak memasukkan anggota kelompok sebelumnya ke dalam daftar *black list*.

```
def kelayakan(NIM_baru):
    print("ini daftar black_list global sekarang: ", black_list)
    print("ini daftar black_list dari", NIM_baru, ":", getBlackList(NIM_baru))
    if str(NIM_baru) in black_list:
        print('gak')
        return False
    if(not isinstance(getBlackList(NIM_baru), list)):
        print('layak')
        return True
    for blackList_baru in getBlackList(NIM_baru):
        if str(blackList_baru) in terpilih:
            print('layak')
            return False
    print('layak')
    return True
```

4. Membuat fungsi seleksi untuk menyeleksi mahasiswa paling kompeten untuk masuk ke kelompok belajar. Fungsi seleksi akan membentuk kelompok dengan jumlah anggota sesuai dengan jumlah anggota yang dimasukkan sebagai parameter fungsi.

```
def seleksi(banyakAnggota):
    counter= 0
    for i in range(banyakAnggota):
        layak = False
        while not layak and counter < 12:
            print("layak kah? ", toCheck["NIM"].iloc[counter])
            layak = kelayakan(toCheck["NIM"].iloc[counter])
            if(layak):
                terpilih.append(toCheck["NIM"].iloc[counter])
                if(isinstance(getBlackList(toCheck["NIM"].iloc[counter]), list)):
                    for new in getBlackList(toCheck["NIM"].iloc[counter]):
                        if str(new) not in black_list:
                            black_list.append(new)
            layak = True
            counter += 1
        if(len(terpilih) != banyakAnggota + 1):
            print("Jumlah anggota yang diinginkan tak terpenuhi")
```

C. Pengujian

Untuk menguji algoritma *greedy* yang telah dibuat di atas, dilakukan dua buah pengujian dengan beberapa data.

1. Pengujian pertama

Berikut ini adalah data mahasiswa yang dijadikan sebagai data masukan:

	NIM	NAMA	NILAI_KUIS	NILAI_TUGAS1	NILAI_TUGAS2	NILAI_TUGAS3	NILAI_UTS	SUDAH_PUNYA_KELompok	BLACK_LIST	NILAI_RATAAN_TUGAS
0	13521001	Ani	64	31	46	75	60	False	13521015 13521012	50.666667
1	13521002	Budi	89	78	89	92	68	False	13521004 13521006	86.333333
2	13521003	Siti	45	50	53	58	50	False	13521013	53.666667
3	13521004	Kumala	77	80	83	84	97	True	13521012	82.333333
4	13521005	Kash	96	88	79	67	70	False	13521004 13521005	78.000000
5	13521006	Tedy	68	62	72	72	84	True	13521001 13521002 13521003	68.666667
6	13521007	William	88	83	83	78	92	False	13521009 13521014	81.333333
7	13521008	Matthew	76	66	69	94	69	False	13521006	83.000000
8	13521009	Lisa	92	95	88	83	80	False	13521004 13521012 13521013	88.666667
9	13521010	Gita	82	93	83	72	75	False	13521001	82.666667
10	13521011	Feb	79	85	76	82	84	False	13521004	81.000000
11	13521012	Hary	76	78	89	78	90	True	NaN	81.666667
12	13521013	Dida	100	82	93	83	94	False	13521009 13521001	85.000000
13	13521014	Michael	74	79	86	93	81	False	NaN	85.000000
14	13521015	Bruno	69	66	82	82	79	False	13521003 13521004	76.666667

Setelah menyembunyikan data mahasiswa yang sudah mempunyai kelompok dan mengurutkan data mahasiswa berdasarkan nilai rata-rata tugas besar, didapatkan tabel data sebagai berikut

	NIM	NAMA	NILAI_KUIS	NILAI_TUGAS1	NILAI_TUGAS2	NILAI_TUGAS3	NILAI_UTS	SUDAH_PUNYA_KELompok	BLACK_LIST	NILAI_RATAAN_TUGAS
0	13521009	Lisa	92	95	88	83	80	False	13521004 13521012 13521013	88.666667
1	13521002	Budi	89	78	89	92	68	False	13521004 13521006	86.333333
12	13521013	Dida	100	82	93	83	94	False	13521009 13521001	85.000000
13	13521014	Michael	74	79	86	93	81	False	NaN	85.000000
7	13521008	Matthew	76	66	69	94	69	False	13521006	83.000000
8	13521010	Gita	82	93	83	72	75	False	13521001	82.666667
6	13521007	William	88	83	83	78	92	False	13521009 13521014	81.333333
10	13521011	Feb	79	85	76	82	84	False	13521004	81.000000
4	13521005	Kash	96	88	79	67	70	False	13521004 13521005	78.000000
14	13521015	Bruno	69	66	82	82	79	False	13521003 13521004	76.666667
2	13521003	Siti	45	50	53	58	50	False	13521013	53.666667
9	13521001	Ani	64	31	46	75	60	False	13521015 13521012	50.666667

Nilai NaN pada tabel menyatakan bahwa mahasiswa tersebut tidak memiliki orang yang tidak ia sukai.

Berikutnya, dengan menggunakan fungsi-fungsi yang telah kita buat sebelumnya, kita akan mencoba membuat kelompok dengan beranggotakan 5 orang.

```
# Misalkan saja NIM kita adalah 13521155
# NIM pertama yang dimasukkan ke dalam kelompok tentunya adalah NIM kita
black_list = []
terpilih = []
NIM_SAYA = 13521155
terpilih.append(NIM_SAYA)

# Misalkan pula kita tidak memiliki orang yang kita tidak sukai
# Sehingga tak perlu menambahkan NIM ke arr black_list awal

# Memilih 4 anggota lainnya untuk kelompok kita
seleksi(4)

print("Berikut ini adalah kelompok yang terbentuk:")
print(terpilih)
print("Rataan Nilai Kelompok:", str(rataan_kelompok(terpilih, NIM_SAYA)))
```

Dengan kode di atas, kita mengasumsikan bahwa NIM pembuat kelompok adalah 13521155 dan pembuat kelompok tersebut tidak memiliki orang yang tidak ia sukai. Dengan kode tersebut, ia akan memilih 4 mahasiswa lainnya untuk menjadi anggota kelompoknya. Berdasarkan kode di atas, keluaran yang akan dihasilkan adalah sebagai berikut:

```

layak kah? 13521009
ini daftar black_list global sekarang: []
ini daftar black_list dari 13521009 : ['13521004', '13521012', '13521013']
layak
layak kah? 13521002
ini daftar black_list global sekarang: [['13521004', '13521012', '13521013']]
ini daftar black_list dari 13521002 : ['13521004', '13521006']
layak
layak kah? 13521013
ini daftar black_list global sekarang: [['13521004', '13521012', '13521013', '13521006']]
ini daftar black_list dari 13521013 : ['13521009', '13521001']
gak
layak kah? 13521014
ini daftar black_list global sekarang: [['13521004', '13521012', '13521013', '13521006']]
ini daftar black_list dari 13521014 : nan
layak
layak kah? 13521008
ini daftar black_list global sekarang: [['13521004', '13521012', '13521013', '13521006']]
ini daftar black_list dari 13521008 : ['13521006']
layak
Berikut ini adalah kelompok yang terbentuk:
[13521155, 13521009, 13521002, 13521014, 13521008]
hai [13521155, 13521009, 13521002, 13521014, 13521008]
Rataan Nilai Kelompok: 86.0

```

Berdasarkan hasil keluaran di atas, algoritma *greedy* berhasil menemukan kombinasi anggota kelompok dengan kompetensi terbaik, yaitu dapat terlihat dari rataan nilai tubes untuk satu kelompok tersebut adalah maksimal, yaitu 86.0 (nilai rataan dari 4 anggota lainnya, tidak termasuk nilai dari pembuat kelompok)

2. Pengujian kedua

Walaupun pada pengujian pertama didapatkan hasil yang optimal dengan menggunakan algoritma *greedy* tersebut, ternyata terdapat kasus dimana algoritma *greedy* yang telah penulis tentukan di atas tidak bekerja optimal. Dengan hanya menambahkan data `black_list` pada `record` dengan nama lisa pada tabel sebelumnya, akan didapatkan data tabel sebagai berikut:

ID	NAMA	NIKELAI_001	NIKELAI_002	NIKELAI_003	NIKELAI_004	NIKELAI_005	NIKELAI_006	NIKELAI_007	NIKELAI_008	NIKELAI_009	NIKELAI_010	NIKELAI_011	NIKELAI_012	NIKELAI_013	NIKELAI_014	NIKELAI_015	NIKELAI_016	NIKELAI_017	NIKELAI_018	NIKELAI_019	NIKELAI_020	NIKELAI_021	NIKELAI_022	NIKELAI_023	NIKELAI_024	NIKELAI_025	NIKELAI_026	NIKELAI_027	NIKELAI_028	NIKELAI_029	NIKELAI_030	NIKELAI_031	NIKELAI_032	NIKELAI_033	NIKELAI_034	NIKELAI_035	NIKELAI_036	NIKELAI_037	NIKELAI_038	NIKELAI_039	NIKELAI_040	NIKELAI_041	NIKELAI_042	NIKELAI_043	NIKELAI_044	NIKELAI_045	NIKELAI_046	NIKELAI_047	NIKELAI_048	NIKELAI_049	NIKELAI_050	NIKELAI_051	NIKELAI_052	NIKELAI_053	NIKELAI_054	NIKELAI_055	NIKELAI_056	NIKELAI_057	NIKELAI_058	NIKELAI_059	NIKELAI_060	NIKELAI_061	NIKELAI_062	NIKELAI_063	NIKELAI_064	NIKELAI_065	NIKELAI_066	NIKELAI_067	NIKELAI_068	NIKELAI_069	NIKELAI_070	NIKELAI_071	NIKELAI_072	NIKELAI_073	NIKELAI_074	NIKELAI_075	NIKELAI_076	NIKELAI_077	NIKELAI_078	NIKELAI_079	NIKELAI_080	NIKELAI_081	NIKELAI_082	NIKELAI_083	NIKELAI_084	NIKELAI_085	NIKELAI_086	NIKELAI_087	NIKELAI_088	NIKELAI_089	NIKELAI_090	NIKELAI_091	NIKELAI_092	NIKELAI_093	NIKELAI_094	NIKELAI_095	NIKELAI_096	NIKELAI_097	NIKELAI_098	NIKELAI_099	NIKELAI_100
0	13521001	Ari	64	31	46	75	62	False	13521015	13521012	50.666667																																																																																										
1	13521002	Budi	89	76	89	92	68	False	13521004	13521008	86.333333																																																																																										
2	13521003	Siti	45	50	53	58	50	False	13521013	50.666667																																																																																											
3	13521004	Karna	77	80	83	84	97	True	13521012	82.333333																																																																																											
4	13521005	Kasih	96	88	79	67	70	False	13521004	13521008	78.000000																																																																																										
5	13521006	Teddy	68	62	72	72	84	True	13521001	13521003	68.666667																																																																																										
6	13521007	William	88	83	83	78	92	False	13521009	13521014	81.333333																																																																																										
7	13521008	Matthew	76	66	89	94	69	False	13521006	83.000000																																																																																											
8	13521009	Lisa	92	95	88	83	80	False	13521002	13521014	13521008	13521012	86.666667																																																																																								
9	13521010	Gita	82	93	83	72	75	False	13521011	81.666667																																																																																											
10	13521011	Febi	79	85	76	82	84	False	13521004	81.000000																																																																																											
11	13521012	Harry	76	78	89	78	90	True	nan	81.666667																																																																																											
12	13521013	Dinda	100	82	93	83	94	False	13521009	13521011	86.000000																																																																																										
13	13521014	Michael	74	79	86	93	81	False	nan	86.000000																																																																																											
14	13521015	Bruno	69	66	82	82	79	False	13521003	13521004	76.666667																																																																																										

Setelah melewati tahap pemrosesan, akan didapati tabel dengan data terurut sebagai berikut

ID	NAMA	NIKELAI_001	NIKELAI_002	NIKELAI_003	NIKELAI_004	NIKELAI_005	NIKELAI_006	NIKELAI_007	NIKELAI_008	NIKELAI_009	NIKELAI_010	NIKELAI_011	NIKELAI_012	NIKELAI_013	NIKELAI_014	NIKELAI_015	NIKELAI_016	NIKELAI_017	NIKELAI_018	NIKELAI_019	NIKELAI_020	NIKELAI_021	NIKELAI_022	NIKELAI_023	NIKELAI_024	NIKELAI_025	NIKELAI_026	NIKELAI_027	NIKELAI_028	NIKELAI_029	NIKELAI_030	NIKELAI_031	NIKELAI_032	NIKELAI_033	NIKELAI_034	NIKELAI_035	NIKELAI_036	NIKELAI_037	NIKELAI_038	NIKELAI_039	NIKELAI_040	NIKELAI_041	NIKELAI_042	NIKELAI_043	NIKELAI_044	NIKELAI_045	NIKELAI_046	NIKELAI_047	NIKELAI_048	NIKELAI_049	NIKELAI_050	NIKELAI_051	NIKELAI_052	NIKELAI_053	NIKELAI_054	NIKELAI_055	NIKELAI_056	NIKELAI_057	NIKELAI_058	NIKELAI_059	NIKELAI_060	NIKELAI_061	NIKELAI_062	NIKELAI_063	NIKELAI_064	NIKELAI_065	NIKELAI_066	NIKELAI_067	NIKELAI_068	NIKELAI_069	NIKELAI_070	NIKELAI_071	NIKELAI_072	NIKELAI_073	NIKELAI_074	NIKELAI_075	NIKELAI_076	NIKELAI_077	NIKELAI_078	NIKELAI_079	NIKELAI_080	NIKELAI_081	NIKELAI_082	NIKELAI_083	NIKELAI_084	NIKELAI_085	NIKELAI_086	NIKELAI_087	NIKELAI_088	NIKELAI_089	NIKELAI_090	NIKELAI_091	NIKELAI_092	NIKELAI_093	NIKELAI_094	NIKELAI_095	NIKELAI_096	NIKELAI_097	NIKELAI_098	NIKELAI_099	NIKELAI_100
0	13521009	Lisa	92	95	88	83	80	False	13521002	13521014	13521008	13521012	86.666667																																																																																								
1	13521002	Budi	89	76	89	92	68	False	13521004	13521006	86.333333																																																																																										
12	13521013	Dinda	100	82	93	83	94	False	13521009	13521011	86.000000																																																																																										
13	13521014	Michael	74	79	86	93	81	False	nan	86.000000																																																																																											
7	13521008	Matthew	76	66	89	94	69	False	13521006	83.000000																																																																																											
9	13521010	Gita	82	93	83	72	75	False	13521011	81.666667																																																																																											
6	13521007	William	88	83	83	78	92	False	13521009	13521014	81.333333																																																																																										
10	13521011	Febi	79	85	76	82	84	False	13521004	81.000000																																																																																											
4	13521005	Kasih	96	88	79	67	70	False	13521004	13521008	78.000000																																																																																										
14	13521015	Bruno	69	66	82	82	79	False	13521003	13521004	76.666667																																																																																										
2	13521003	Siti	45	50	53	58	50	False	13521013	50.666667																																																																																											
8	13521001	Ari	64	31	46	75	62	False	13521015	13521012	50.666667																																																																																										

Dengan menggunakan kode pengujian yang sama dengan kode pada pengujian pertama, didapatkan hasil keluaran sebagai berikut.

```

layak kah? 13521009
ini daftar black_list global sekarang: []
ini daftar black_list dari 13521009 : ['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']
layak
layak kah? 13521002
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521002 : ['13521004', '13521006']
layak
layak kah? 13521013
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521013 : ['13521009', '13521001']
layak
layak kah? 13521014
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521014 : nan
layak
layak kah? 13521008
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521008 : ['13521006']
layak
layak kah? 13521013
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521013 : ['13521009', '13521001']
layak
layak kah? 13521014
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521014 : nan
layak
layak kah? 13521008
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521008 : ['13521006']
layak
layak kah? 13521013
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521013 : ['13521009', '13521001']
layak
layak kah? 13521014
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521014 : nan
layak
layak kah? 13521008
ini daftar black_list global sekarang: [['13521002', '13521004', '13521008', '13521012', '13521013', '13521007', '13521006', '13521011']]
ini daftar black_list dari 13521008 : ['13521006']
layak
Berikut ini adalah kelompok yang terbentuk:
[13521155, 13521009, 13521002, 13521014, 13521008]
hai [13521155, 13521009, 13521002, 13521014, 13521008]
Rataan Nilai Kelompok: 73.50000000000001

```

Berdasarkan hasil keluaran tersebut, algoritma *greedy* tidak menghasilkan kombinasi anggota kelompok dengan kompetensi terbaik karena komposisi anggota kelompok terbaik yang mungkin dibuat adalah [13521155, 13521002, 13521013, 13521014, 13521008] dengan rataan nilai tubes 4 mahasiswa (selain pembuat kelompok) adalah 85.33. Hal ini membuktikan bahwa, algoritma *greedy* tidak memberikan hasil yang optimal apabila terdapat anggota dengan kompetensi tinggi yang tidak menyukai banyak orang ataupun tidak disukai banyak orang.

IV. PENUTUP

A. KESIMPULAN

Algoritma *Greedy* yang telah ditetapkan di atas tidak selalu dapat digunakan untuk membentuk kelompok dengan kompetensi terbaik. Algoritma *Greedy* tak efektif digunakan untuk persoalan pembentukan kelompok apabila terdapat mahasiswa yang memiliki kompetensi tinggi (nilai rata-rata yang tinggi), namun tak disukai oleh banyak orang. Algoritma ini juga tidak efektif apabila terdapat mahasiswa dengan kompetensi tinggi yang tidak menyukai banyak orang. Dengan memilih mahasiswa tersebut terlebih dahulu menjadi anggota kelompok, mahasiswa lainnya tak akan memenuhi persyaratan (memenuhi fungsi kelayakan) untuk menjadi anggota kelompok.

B. SARAN

Algoritma *Greedy* yang ada dapat disempurnakan kembali, misalnya dengan memberikan pembobotan nilai ketidaksukaan seseorang dengan orang lain terhadap kemampuan kompetensinya (nilai rata-rata mahasiswa tersebut). Atribut `black_list` yang merupakan daftar orang yang tidak disukai oleh seseorang mungkin juga dapat diganti menjadi skor EQ mahasiswa (kemampuan seseorang untuk menerima, mengelola, dan mengontrol emosi dirinya)

sehingga setiap atribut data mahasiswa yang kita perhitungkan merupakan atribut yang dapat diukur sehingga akan mempermudah kita untuk memberikan pembobotan nilai kompetensi dan kemampuan sosialisasi mahasiswa tersebut.

VIDEO LINK AT YOUTUBE

<https://youtu.be/Esm1GWIw1g>

REFERENSI

- [1] <http://lib.unnes.ac.id/28634/1/1511412014.pdf>
- [2] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag1.pdf)
- [3] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Greedy-(2021)-Bag2.pdf)
- [4] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-\(2022\)-Bag3.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Greedy-(2022)-Bag3.pdf)

- [5] <https://www.detik.com/edu/detikpedia/d-5988134/apa-itu-kompetensi-ini-pengertian-faktor-faktor-dan-indikatornya>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 20 Mei 2023



Kandida Edgina Gunawan - 13521155