

Implementasi *Tour Planning* pada Trip Perjalanan Mengelilingi Swiss dari 6 *Airport International Swiss*

Samuel Christopher Swandi - 13520075

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13520075@std.stei.itb.ac.id

Abstrak – Pandemi Covid yang sudah mulai mereda membuat aktivitas manusia menjadi kembali seperti normal, tak terkecuali kegiatan berlibur dan berpergian. Karena berpergian merupakan salah satu ajang untuk berelaksasi serta memanjakan mata, maka kebanyakan orang sangat menyeleksi tujuan berpergian mereka sehingga mereka mendapatkan wisata yang terbaik. Salah satu tujuan berpergian terbaik adalah sebuah negara di Eropa yaitu Switzerland, atau disingkat, Swiss. Switzerland menawarkan berbagai wisata yang memanjakan mata wisatawan, dan juga telah dinominasi sebagai salah satu tempat wisata terbaik di dunia berdasarkan majalah Times. Makalah ini membahas tentang perencanaan rute terbaik untuk mengunjungi seluruh tempat wisata di Swiss, dari 6 bandara internasional yang ada di sana.

Kata kunci – Route Planning, Switzerland, Swiss

I. PENDAHULUAN

Pandemi Covid yang sudah mulai mereda menimbulkan banyak orang menjadi lebih sering berpergian. Salah satu tujuan berpergian terbaik berdasarkan majalah Times, adalah Switzerland atau disebut juga Swiss. Switzerland menawarkan berbagai wisata dari pedesaan sampai pegunungan. Sebagai orang yang telah memilih Swiss sebagai negara wisata pilihannya, banyak orang yang juga ingin mengelilingi seluruh wisata yang ada di Swiss.

Makalah kali ini difokuskan untuk membahas rute perjalanan terbaik untuk mengunjungi seluruh tempat wisata terbaik di Swiss, melalui 6 bandara internasional yang terdapat di sana. Untuk penyelesaian perencanaan rute terbaik, penulis menggunakan algoritma branch and bound untuk menentukan rute tercepat ke masing-masing destinasi wisata yang tersedia.

II. LANDASAN TEORI

A. Switzerland

Swiss, secara resmi Konfederasi Swiss, adalah negara yang terkurung di antara Eropa Barat, Tengah dan Selatan. Negara ini adalah republik federal yang terdiri dari 26 kanton, dengan otoritas federal yang berbasis di Bern. Swiss berbatasan dengan Italia di selatan, Prancis di barat, Jerman di utara, serta Austria dan Liechtenstein di timur. Secara geografis dibagi di antara

Dataran Tinggi Swiss, Pegunungan Alpen dan Jura, dengan luas total 41.285 km² (15.940 sq mi) dan luas daratan 39.997 km² (15.443 sq mi). Meskipun Pegunungan Alpen menempati sebagian besar wilayah, populasi Swiss sekitar 8,5 juta sebagian besar terkonsentrasi di dataran tinggi, di mana kota-kota terbesar dan pusat-pusat ekonomi, di antaranya Zürich, Geneva dan Basel. Ketiga kota ini merupakan rumah bagi beberapa kantor organisasi internasional seperti WTO, WHO, ILO, markas besar FIFA, kantor terbesar kedua PBB, serta kantor utama Bank for International Settlements. Bandara internasional utama Swiss juga terletak di kota-kota ini.

Enam bandara internasional terbesar di Swiss antara lain:

1. Zurich Airport
2. Geneva Airport
3. Basel Airport
4. St. Gallen Airport
5. Lugano Airport
6. Bern Airport

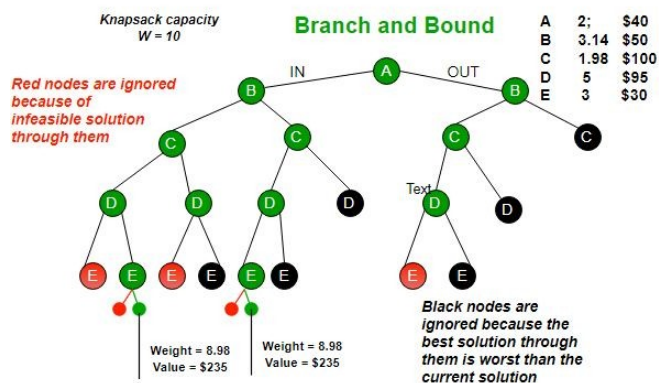
Keenam bandara internasional di atas merupakan 6 bandara terbesar di Swiss, dan merupakan tempat untuk berbagai wisatawan mendarat. Sehingga penulis memakai keenam bandara tersebut menjadi titik mulai untuk perjalanan wisata ke seluruh swiss.



Gambar 1.0 Tempat wisata terbaik di Swiss, berdasarkan peta
sumber: <https://www.google.com/>

1. Basel
2. Zurich
3. St Gall
4. Chur
5. Klosters
6. Davos
7. St Moritz
8. Lugano
9. Lucerne
10. Bern
11. Interlaken
12. Glinderwald
13. Bern
14. Neuchatel
15. Biel
16. Lausanne
17. Geneva
18. Verbier
19. Martigny
20. Zermatt

B. Branch and Bound



Algoritma Branch and Bound (BnB) adalah salah satu algoritma penyelesaian soal optimasi. Persoalan optimasi adalah persoalan untuk mencari nilai minimum ataupun maksimum dari suatu objektif yang dideskripsikan pada soal. Contoh soal yang sering diselesaikan dengan algoritma bnb antara lain, *shortest path in a graph* dan *minimum spanning tree*, dan algoritma bnb berhasil menyelesaikannya dengan waktu polinomial.

1. Branching

2. Bounding

Algoritma bnb ini mirip dengan algoritma *Breadth First Search (BFS)* dengan *Least Cost Search (LCS)*. Perbedaan antara 2 algoritma tersebut terletak pada “bound” di mana *bounding* membuat algoritma menjadi lebih efisien karena memangkas sebagian besar jalur.

1. Didefinisikan tabel graf berupa matriks dua dimensi yang mendefinisikan jarak antara simpul
2. Untuk setiap simpul, diberikan rumus $cost(i)$, di mana $cost(i)$ adalah sebuah nilai heuristik untuk menghitung biaya (dalam hal ini, penulis menganggap biaya = jarak), dari simpul I ke simpul tujuan
3. Membangkitkan simpul - simpul baru yang bertetangga dengan simpul I, lalu menghitung $cost(i)$ untuk masing-masing simpul yang dibangkitkan.
4. Jika simpul tujuan sudah ditemukan, hapus seluruh simpul aktif yang mempunyai biaya lebih besar dari simpul tujuan sekarang. Jika masih ada simpul yang aktif, maka tetap ditelusuri lebih lanjut
5. Ulangi tahap ke 3 sampai tidak ada lagi simpul yang aktif.

A map of Europe with a network of red lines and dots representing connections between various locations. The dots are placed at major cities or hubs, and the lines connect them, forming a complex web across the continent.

Travelling Salesman Problem adalah sebuah persoalan matematika yang menanyakan “Diberi sekumpulan kota dan jarak antar tiap kota, manakah rute tercepat/tersingkat untuk mengunjungi seluruh kota dengan hanya satu kali mengunjungi dan kembali ke kota asal?”

Masalah ini sering sekali dijadikan tes untuk banyak sekali algoritma optimasi. Banyak sekali solusi untuk persoalan TSP. Sebagai contoh, TSP dapat diselesaikan menggunakan exhaustive search algorithm yang menelusuri seluruh kota dengan seluruh kombinasi yang ada untuk menentukan rute tercepatnya. Metode exhaustive search digaransikan akan memberikan solusi terbaik, tapi jika kota yang disediakan tambah banyak, maka kombinasi yang muncul juga akan bertambah banyak, sehingga waktu komputasi yang digunakan juga akan sangat banyak! Dalam kompleksitas waktu yang ditranslasikan dalam notasi big-Oh, exhaustive search untuk persoalan TSP mendapat kompleksitas sebesar $O(n!)$, dengan n merupakan banyaknya kota yang tersedia. Sehingga, dalam makalah ini, penulis akan menggunakan algoritma yang lebih efisien untuk menyelesaikan persoalan TSP dalam mencari rute tercepat untuk travelling ke seluruh Swiss, yaitu dengan menggunakan algoritma branch and bound.

III. IMPLEMENTASI DAN EKSPERIMEN

Sebelum memulai implementasi, penulis harus melakukan riset untuk jarak antara tiap kota yang ada di Swiss dengan menggunakan titik lintang pada peta. Sehingga didapatkan hasil sebagai berikut

Nama Kota	Lintang	Bujur
Basel	47.5596° N	7.5886° E
Zurich	47.3769° N	8.5417° E
St Gallen	47.4245° N	9.3767° E
Chur	46.8508° N	9.5320° E
Klosters	46.8699° N	9.8829° E
Davos	46.8027° N	9.8360° E
St. Moritz	46.4908° N	9.8355° E
Lugano	46.0037° N	8.9511° E
Lucerne	47.0502° N	8.3093° E
Bern	46.9480° N	7.4474° E
Interlaken	46.6863° N	7.8632° E
Gliderwald	46.6242° N	8.0414° E
Bern	46.6242° N	8.0414° E
Neuchatel	46.9900° N	6.9293° E
Biel	47.1368° N	7.2468° E
Lausanne	46.5197° N	6.6323° E
Geneva	46.2044° N	6.1432° E
Verbier	46.0961° N	7.2286° E
Martigny	46.1050° N	7.0755° E
Zermatt	46.0207° N	7.7491° E

Untuk menghitung jarak antara tiap kota, penulis melakukan riset lagi tentang jarak antara dua titik lintang berdasarkan derajat, dan didapatkan bahwa 1 derajat perbedaan setara dengan 111 Km. Sehingga dapat dirumuskan jarak antara tiap kota adalah:

$$(111 \text{ km/derajat}) * \sqrt{(LU1 - LU2)^2 + (LT1 - LT2)^2}$$

$LU1$ = Lintang Utara Kota Asal
 $LU2$ = Lintang Utara Kota Tujuan
 $LT1$ = Lintang Timur Kota Asal
 $LT2$ = Lintang Timur Kota Tujuan

Sehingga, dari rumus yang telah diterapkan di atas, penulis dapat membuat sebuah graf hubungan antara tiap kota di Swiss. Berikut implementasi dalam bahasa pemrograman python:

1. Data struktur destinasi

```
# Representasi LONGITUDE and LATITUDE
class Coordinate:
    def __init__(self, long, lat):
        self.long = long
        self.lat = lat

    def show(self):
        print(f"{self.long}°N, {self.lat}°E ")

# Representasi nama destinasi, dan koordinat di peta
class Destination:
    def __init__(self, name, long, lat, time=0):
        self.name = name
        self.coordinate = Coordinate(long, lat)
        self.time = time

    def show(self):
        print(f"{self.name}" + " " * (12-len(self.name)), end="")
        self.coordinate.show()
```

Gambar 3.0 struktur data koordinat dan destinasi wisata dalam python
sumber: dokumen pribadi penulis

2. Perhitungan jarak graf

```
adj = []
for i in listOfDestination:
    temp = []
    for j in listOfDestination:
        # jarak untuk 1 derajat perbedaan longitude dan latitude adalah 111 km
        dist = round(distanceBetween(i.coordinate, j.coordinate)*111, 2)
        if dist == 0:
            temp.append(0)
        else:
            temp.append(dist)
    adj.append(temp)
```

Gambar 3.1 perhitungan graf
sumber: dokumen pribadi penulis

```
# Jarak antara koordinat a and b
def distanceBetween(a, b):
    return round(math.sqrt((a.long - b.long)**2 + (a.lat - b.lat)**2), 2)
```

Gambar 3.2 perhitungan jarak antara 2 kota
sumber: dokumen pribadi penulis

Dan dihasilkan graf sebagai berikut:

Gambar 3.3 graf jarak antar kota
sumber: dokumen pribadi penulis

Baris berarti kota asal, dan kolom berarti jarak ke kota tujuan, dan kota diurutkan berdasarkan tabel di atas yang sudah penulis deskripsikan di atas. Sebagai contoh, untuk baris pertama, berarti kota asal adalah Zurich, dan sisanya ke kota lainnya, maka dapat dibuat tabel sebagai berikut:

Kota Tujuan	Jarak (km)
Zurich	0
Geneva	296,37
Basel	107,67
Bern	130,98
Lugano	158,73
St. Gallen	93,24
Chur	124,32
Klosters	158,73
Davos	157,62
St. Moritz	174,27
Lucerne	44,4
Interlaken	107,67
GlinderWald	99,9
Neuchatel	184,26
Biel	146,52
Lausanne	231,99
Verbier	203,13
Martigny	215,34
Zermatt	174,27

Nol pada kota Zurich, berarti kota tersebut merupakan kota yang sedang dikunjungi.

A. Implementasi Branch and Bound

Pada implementasinya, penulis memisah beberapa fungsi agar menjadi lebih modular. Sehingga fungsi yang didefinisikan adalah:

1. Copy To Final

Bertugas untuk sementara menyalin hasil dari *path* sekarang ke jawaban akhir (final)

2. First Min

Bertugas untuk mencatat jarak paling minimum dalam graf

3. Second Min

Bertugas untuk mencatat jarak kedua paling minimum pada graf

4. TSP Recursive

Bertugas sebagai algoritma utama untuk implementasi penyelesaian persoalan tour planning dengan branch and bound

5. TSP

Bertugas untuk mencatat *bound* awal, dan meneruskan pilihan kota asal dari user ke TSP recursive untuk diproses lebih lanjut.

6. Main Program

Bertugas sebagai jalur input dan output dari user dan program, dan bertujuan untuk menerima masukan user berupa bandara asal untuk memulai perjalanan keliling Swiss.

Detail tahap penyelesaian branch and bound pada TSP Recursive, beserta implementasinya pada python:

1. Membuat kondisi basis untuk rekursi selesai, yaitu jika level atau kedalaman dari pohon yang sudah dibuat sama dengan panjang kota yang ingin dikunjungi (19 kota).
2. Jika tidak masuk ke kondisi basis, maka akan memulai iterasi dari 0 sampai N dengan N banyaknya kota
3. Mengecek jika kota tersebut sudah dikunjungi atau belum, jika sudah maka akan dilanjutkan untuk mengecek nilai minimum dari kolom dan baris
4. Jika sudah menemukan nilai minimum dari kolom dan baris, maka akan dihitung cost, jika cost lebih kecil dari *final_size*, maka akan dilanjutkan rekursi untuk level berikutnya. Rekursi untuk level berikutnya di mulai kembali dari poin 1
5. Jika rekursi sudah selesai, maka akan dicatat hasil akhir dari perjalanan rute ke dalam variabel bernama *final_path* sehingga program dapat melihat runtutan perjalanan dari variabel tersebut

B. Test Case

1. Zurich Airport

Jika mulai dari Zurich Airport, maka perjalanan yang akan dilakukan adalah:

```
Pilihan airport untuk memulai menjelajahi seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 1

Jarak tempuh : 1136.64 km
Rute perjalanan : Zurich -> Lucerne -> Glinderwald -> Interlaken -> Bern -> Basel -> Biel -> Neuchatel -> Lausanne -> Geneva -> Martigny -> Verbier -> Zermatt -> Lugano -> St Moritz -> Davos -> Klosters -> Chur -> St Gallen -> Zurich
```

Gambar 3.4 hasil komputasi rute perjalanan dari kota Zurich
sumber: dokumen pribadi penulis

Dengan total perjalanan 1136,64 Km, dan jika diilustrasikan dalam peta, maka terdapat hasil sebagai berikut:



Gambar 3.3 graf jarak antar kota
sumber: dokumen pribadi penulis

Dapat dilihat di ilustrasi di atas, jika kita mendarat dari bandara mana pun, maka jalur yang akan dilalui akan selalu sama

2. Geneva Airport

Jika mendarat di Geneva Airport, maka jalur yang harus dilewati adalah:

```
Pilihan airport untuk memulai menjelajahi seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 2
```

Makalah IF2211 Strategi Algoritma, Semester II Tahun 2021/2022

```
Jarak tempuh : 1136.64 km
Rute perjalanan : Geneva -> Lausanne -> Neuchatel -> Biel -> Basel -> Bern -> Interlaken -> Glinderwald -> Lucerne -> Zurich -> St Gallen -> Chur -> Klosters -> Davos -> St Moritz -> Lugano -> Zermatt -> Verbier -> Martigny -> Geneva
```

Gambar 3.5 hasil komputasi rute perjalanan dari kota Geneva
sumber: dokumen pribadi penulis

3. Basel Airport

Jika mendarat di Basel Airport, maka jalur yang harus dilewati adalah:

```
Pilihan airport untuk memulai menjelajahi seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 3

Jarak tempuh : 1136.64 km
Rute perjalanan : Basel -> Bern -> Interlaken -> Glinderwald -> Lucerne -> Zurich -> St Gallen -> Chur -> Klosters -> Davos -> St Moritz -> Lugano -> Zermatt -> Verbier -> Martigny -> Geneva -> Lausanne -> Neuchatel -> Biel -> Basel
```

Gambar 3.6 hasil komputasi rute perjalanan dari kota Basel
sumber: dokumen pribadi penulis

4. Bern Airport

Jika mendarat di Bern Airport, maka jalur yang harus dilewati adalah:

```
Pilihan airport untuk memulai menjelajahi seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 4

Jarak tempuh : 1136.64 km
Rute perjalanan : Bern -> Interlaken -> Glinderwald -> Lucerne -> Zurich -> St Gallen -> Chur -> Klosters -> Davos -> St Moritz -> Lugano -> Zermatt -> Verbier -> Martigny -> Geneva -> Lausanne -> Neuchatel -> Biel -> Basel -> Bern
```

Gambar 3.7 hasil komputasi rute perjalanan dari kota Bern

5. Lugano Airport

Jika mendarat di Lugano Airport, maka jalur yang harus dilewati adalah:

```
Pilihan airport untuk memulai menjelajahi
seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 5

Jarak tempuh : 1136.64 km
Rute perjalanan : Lugano -> Zermatt -> Ver
bier -> Martigny -> Geneva -> Lausanne ->
Neuchatel -> Biel -> Basel -> Bern -> Inte
rlaken -> Glinderwald -> Lucerne -> Zurich
-> St Gallen -> Chur -> Klosters -> Davos
-> St Moritz -> Lugano
```

Gambar 3.8 hasil komputasi rute perjalanan dari kota Lugano
sumber: dokumen pribadi penulis

6. St. Gallen Airport

Jika mendarat di St. Gallen Airport, maka jalur yang harus dilewati adalah:

```
Pilihan airport untuk memulai menjelajahi
seluruh swiss:
1. Zurich
2. Geneva
3. Basel
4. Bern
5. Lugano
6. St Gallen

Pilihan airport untuk mulai (angka): 6

Jarak tempuh : 1139.97 km
Rute perjalanan : St Gallen -> Zurich -> L
ucerne -> Glinderwald -> Interlaken -> Ber
n -> Basel -> Biel -> Neuchatel -> Lausann
e -> Geneva -> Martigny -> Verbier -> Zerm
att -> Lugano -> St Moritz -> Klosters ->
Davos -> Chur -> St Gallen
```

Gambar 3.9 hasil komputasi rute perjalanan dari kota St. Gallen
sumber: dokumen pribadi penulis

Dari keenam bandara di atas, dapat dilihat bahwa jarak yang ditempuh untuk melewati seluruh Swiss adalah sama, sehingga jika ingin mendarat dari manapun, kita dapat dengan mudah menelusuri seluruh Swiss melalui rute yang sudah diberikan.

Namun, kita dapat menentukan rute terbaik untuk dilalui berdasarkan wisata-wisata yang dilalui dan keruntutan pemandangan yang dilewati selama perjalanan. Dalam hal ini, pilihan terbaik untuk mendarat adalah bandara Zurich. Dikarenakan Kota Zurich memiliki rute langsung ke kota Lucerne, Interlaken, Grindelwald, Bern, dan Basel dalam sekali perjalanan dan sekaligus merupakan bandara internasional terbesar milik Switzerland. Kota Interlaken, Grindelwald dan

Bern merupakan kota terbaik yang menjadi tempat tujuan wisatawan yang ingin berkunjung ke Swiss, sehingga jika melewati bandara Zurich, kita dapat langsung melakukan perjalanan ke 4 kota tersebut. Dan jika kita tidak memiliki *budget* yang cukup untuk mengelilingi Swiss, maka setelah dari kota Basel, kita dapat kembali langsung ke Zurich. Hal ini dapat dibuktikan dengan melihat rencana tur yang disediakan oleh *tour planner* ternama seperti Golden Rama, ATS Vacation, dan Panorama.

1. Golden Rama Tour

Hari 1: JAKARTA – ZURICH

Hari 2: ZURICH

Hari 3: ZURICH – ST. MORITZ – LUGANO

Hari 4: LUGANO – TAESCH – ZERMATT/BRIG

Hari 5: ZERMATT/BRIG – MONTREUX – INTERLAKEN /GRINDELWALD

Hari 6: INTERLAKEN/GRINDELWALD – JUNGFRAUJOCH – INTERLAKEN/GRINDELWALD

Hari 7: INTERLAKEN/GRINDELWALD – LUCERNE – ZURICH

Hari 8: ZURICH – SCHAFFHAUSEN – ZURICH

Hari 9: ZURICH – JAKARTA

Hari 10: JAKARTA

Gambar 3.10 Jadwal tour golden rama
sumber: golden rama tour

2. ATS Vacation

HARI 1 : JAKARTA – ZURICH

WED, 03 NOV 2021

Hari ini Anda berkumpul di Bandara Soekarno - Hatta untuk menuju Zurich.

HARI 2 : TIBA DI ZURICH – BERN – BIEL

THU, 04 NOV 2021

Setibanya di Zurich Anda akan diantar menuju menuju ibukota negara Swiss yaitu k satu penangkaran Beruang yang dilindungi oleh pemerintah Swiss, jika Anda berun diajak untuk berjalan kaki di sepanjang kota tua Bern sambil berbelanja souvenir kha hotel untuk beristirahat.

Akomodasi: Mercure Plaza Biel hotel atau setaraf

HARI 3 : BIEL – GENEVA – LAUSANNE – VEVEY – MONTREUX – MARTIGNY

FRI, 05 NOV 2021

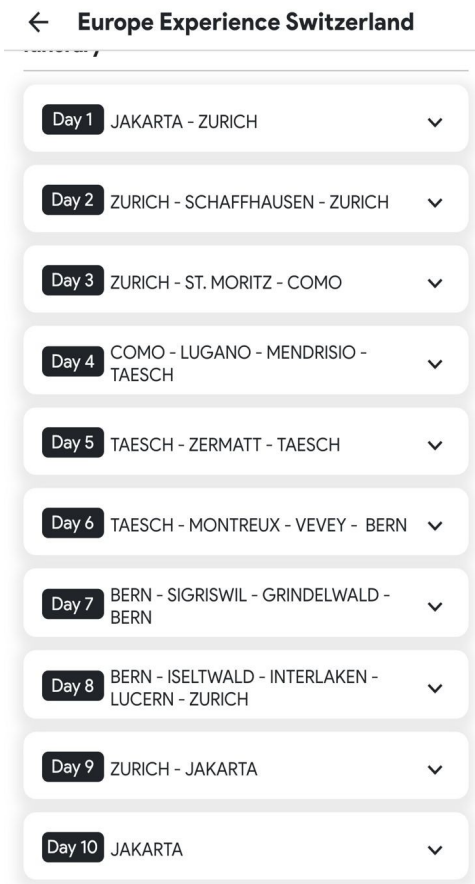
Hari ini anda menuju Geneva yang terkenal dengan danau Geneva yang indah yang berfoto di **Flower Clock** yang indah, serta melewati **Red Cross Building** dan **Palais I Lausanne** dengan melihat **Charlie Chaplin Staue**, **Nestle Palais** yang menjadikan ic Anda akan diajak untuk berfoto di depan **Chillon Castle** yang berada di pinggir dan menakjubkan. Tak lupa berfoto di depan patung **Freddie Mercury** yang melegenda

Akomodasi: Hotel Vatel atau setaraf

☺ Breakfast

Gambar 3.11 Jadwal tour ATS
sumber: ATS Vacation

3. Panorama



Gambar 3.12 Jadwal tour Panorama
sumber: Panorama Tour

IV. KESIMPULAN

Algoritma Branch and Bound dapat digunakan untuk mencari rute terbaik untuk menelusuri seluruh kota yang ada di Swiss. Atau jika masalah ini dijadikan matematis, masalah ini merupakan masalah yang cukup terkenal yaitu *Travelling Salesman Problem*. Algoritma ini memiliki kompleksitas sebesar $O(n^2)$, yang artinya waktu yang dibutuhkan adalah eksponensial.

Menggunakan algoritma Branch and Bound bukan merupakan algoritma terbaik untuk menyelesaikan permasalahan TSP Swiss ini, karena kompleksitas waktu yang

eksponensial. Sehingga waktu komputasi yang dibutuhkan cukup banyak.

Namun, algoritma Branch and Bound ini tetap baik untuk melakukan komputasi untuk TSP dengan simpul yang sedikit, seperti pada makalah ini, hanya terdapat 19 kota yang harus dikunjungi.

REFERENSI

- [1] R. Munir, "Algoritma Branch and Bound", 2022, <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Branch-and-Bound-2021-Bagian1.pdf>
- [2] SCCC, 2022, <https://www.bfs.admin.ch/bfs/de/home/statistiken/mobilitaet-verkehr.assetdetail.14347287.html>
- [3] Panorama JTB Tour, "Tour 10D Jakarta Switzerland", 2022, mobile apps
- [4] Golden Rama Tour, "Tour10D Jakarta Swiss", 2022, <https://m.goldenrama.com/tour/detail/favorite-switzerland-jungfrau-joch-panoramic-train>
- [5] ATS Vacation, "8D Swiss winter wonderland", 2022, <https://www.ats-vacations.com/group-tours/p/08d-swiss-winter-wonderland>
- [6] LatLong.Net, Posisi Lintang dan Bujur kota Swiss", 2022, <https://www.latlong.net/place/predigerkirche-zurich-switzerland-19616.html>

LINK VIDEO MAKALAH

<https://youtu.be/e0jUZ66tIoM>

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 2022

Samuel Christopher Swandi, 13520075