

# Implementasi Algoritma Brute Force Pada Pemilihan Tempat Duduk Bioskop

## *Implementation of Brute Force Algorithm in cinema seating selection*

Aqsal Yazid Setiawan 120140197

Program Studi Teknik Informatika

Jurusan Teknologi Produksi dan Industri

Institut Teknologi Sumatera, Jalan Terusan Ryacudu Lampung

E-mail (gmail): aqsal.120140197@student.itera.ac.id

### *Abstract*

Seiring berkembangnya zaman, bioskop menjadi salah satu pilihan destinasi liburan bagi beberapa kalangan masyarakat. dikarenakan minat pelanggan yang besar untuk menonton sebuah film di bioskop tetapi kapasitas yang tersedia cukup terbatas membuat bioskop menerapkan sistem pemesanan yang memungkinkan pelanggan menentukan tempat duduk mereka jauh sebelum film tersebut tayang sesuai jadwal. maka dari itu, dibutuhkan sebuah algoritma yang dapat menentukan apakah kursi tersebut sudah dipesan atau belum algoritma brute force dapat menjadi sebuah solusi yang dapat menyelesaikan permasalahan tersebut karena, algoritma brute force akan melakukan pengecekan satu per satu dari kursi pertama sampai kursi terakhir. hal ini semua tempat duduk yang tersedia dapat ditentukan statusnya

**Kata Kunci :** *Bioskop, Algoritma Brute Force*

### I. PENDAHULUAN

Bioskop menjadi salah satu destinasi bagi beberapa kalangan untuk menonton film. walaupun beberapa waktu lalu diberlakukan pembatasan bagi beberapa tempat umum, bioskop seperti memiliki peminatnya sendiri. dikarenakan banyaknya konsumen yang ke bioskop, membuat beberapa bioskop membuat sebuah sistem pemesanan yang dapat diakses secara online. sistem tersebut juga harus dapat menampilkan status tempat duduk, apakah sudah dipesan atau masih kosong. untuk menentukan status tersebut, dibutuhkan sebuah algoritma yang dapat melakukan pemeriksaan satu per satu. hal ini menjadikan algoritma brute force menjadi metode terbaik dalam menyelesaikan permasalahan ini. dikarenakan teknik algoritma ini adalah straight forward, membuat algoritma ini akan memecahkan permasalahan ini secara satu per satu sehingga status semua kursi dapat dilihat ketika pelanggan akan melakukan pemesanan

### II. LANDASAN TEORI

#### A. Bioskop

Menurut Wikipedia, Bioskop adalah sebuah tempat untuk menonton pertunjukan film dengan menggunakan layar lebar. Bioskop mejadi salah satu destinasi berlibur oleh beberapa kalangan.



Gambar 2.1 Tempat Duduk Bioskop

Walaupun sekarang banyak situs yang menyediakan layana untuk menonton film secara online dengan sistem berlangganan, tidak membuat bioskop sepi peminatnya. Hal ini membuat beberapa bioskop membuat aplikasi yang dapat diakses secara online untuk melakukan pemesanan tempat duduk serta film yang akan ditonton demi mempermudah konsumen saat ingin menonton film di bioskop



Gambar 2.1 Salah satu aplikasi pemesanan tempat duduk bioskop secara online

### B. Algoritma Brute Force

Algoritma Brute force adalah sebuah pendekatan yang langsung (straightforward) untuk memecahkan suatu masalah, biasanya didasarkan pada pernyataan masalah dan definisi konsep yang dilibatkan.

Algoritma Brute Force tidak cukup cerdas untuk menyelesaikan permasalahan karena membutuhkan volume komputasi yang cukup besar serta waktu yang cukup lama dalam penyelesaian masalah, sehingga algoritma ini tidak menghasilkan algoritma yang efisien.

Kelebihan Algoritma Brute Force :

- Algoritma Brute Force digunakan untuk memecahkan hampir sebagian besar masalah
- Algoritma Brute Force sederhana dan mudah dipahami
- Algoritma Brute Force akan menghasilkan algoritma yang cocok untuk menyelesaikan masalah seperti masalah pencarian status kursi bioskop
- Algoritma Brute Force akan menghasilkan algoritma yang baku untuk tugas-tugas komputasi

Kelemahan Algoritma Brute Force :

- Algoritma brute force jarang menghasilkan algoritma yang efisien
- Algoritma brute force berjalan lambat jika menerima masukan yang berukuran besar
- Algoritma brute force tidak kreatif strategi pemecahan masalah lain untuk masalah tertentu

### C. Sequential Search

Sequential Search adalah algoritma pencarian yang dimana data akan dicari secara berurutan hingga sesuai dengan “key” yang di inputkan.

Sequential Search atau biasa disebut Linear Search termasuk kedalam algoritma pencarian yang paling sederhana yang pada dasarnya algoritma ini akan melakukan proses membandingkan setiap elemen larik secara beruntun dari elemen pertama hingga elemen yang dicari dapat ditemukan dalam larik atau hingga seluruh elemen sudah diperiksa

## III. ANALISIS DAN PEMBAHASAN PERMASALAHAN

Seiring perkembangan teknologi, banyak bioskop yang membuat sebuah sistem pemesanan secara online, sehingga para pelanggan yang akan menonton di bioskop dapat memesan tempat duduk melalui sebuah sistem aplikasi secara online. Sistem tersebut akan menggunakan algoritma brute force untuk menentukan apakah kursi yang akan dipesan masih kosong atau sudah dipesan.

Salah satu metode dari algoritma brute force yang akan digunakan adalah sequential search. Sequential search adalah sebuah teknik pencarian data dimana data dicari secara urut dari depan ke belakang berdasarkan “key” yang dicari.

### A. Sequential Search

Sequential Search dipilih karena algoritma ini akan melakukan pengecekan satu per satu elemen pada elemen larik yang tersedia.

Pada sistem ini akan disediakan larik dan setiap elemen larik memiliki status 0 dan 1, status 0 berarti tempat duduk telah dipesan dan status 1 berarti tempat duduk masih kosong.

Sequential search akan digunakan untuk melakukan pengecekan status dari setiap elemen larik. Semua status akan ditampilkan setelah dilakukan pengecekan.

Sequential Search memiliki pseudocode seperti berikut :

```
procedure PencarianBeruntun(input  $a_1, a_2, \dots, a_n$  : integer,  $x$  : integer; output  $idx$  : integer)
{ Mencari elemen bernilai  $x$  di dalam senarai  $a_1, a_2, \dots, a_n$ . Lokasi (indeks elemen) tempat  $x$ 
ditemukan diisi ke dalam  $idx$ . Jika  $x$  tidak ditemukan, maka  $idx$  diisi dengan 0.
Masukan:  $a_1, a_2, \dots, a_n$ 
Luaran:  $idx$ 
}
Deklarasi
 $k$  : integer

Algoritma:
 $k \leftarrow 1$ 
while ( $k < n$ ) and ( $a_k \neq x$ ) do
 $k \leftarrow k + 1$ 
endwhile
{  $k = n$  or  $a_k = x$  }

if  $a_k = x$  then {  $x$  ditemukan }
 $idx \leftarrow k$ 
else
 $idx \leftarrow -1$  {  $x$  tidak ditemukan }
endif
```

Gambar 3.1 Sequential Search Pseudocode

### B. Python

Python dipilih pada sistem ini karena bahasa python sangat mudah untuk dipahami dan memiliki modul-modul yang mempermudah dalam pembuatan. Modul digunakan pada sistem ini adalah Random dan Numpy



Gambar 3.2 Logo Python

### C. Random

Random adalah salah satu modul pada python yang dapat menginputkan sebuah nilai kepada sebuah variabel secara acak. nilai yang dipakai pada sistem ini adalah 0 dan 1.

Modul Random dapat diakses dengan menuliskan "import random". Modul Random juga memiliki beberapa method atau fungsi, diantara lain :seed, getstate, setstate, getrandbits, randint, dan masih banyak lagi. Setiap method tersebut memiliki fungsi yang berbeda, tetapi di sistem ini hanya menggunakan method randint

### D. Numpy

Numpy adalah salah satu library yang ada pada python. Numpy berfungsi untuk melakukan proses komputasi numerik. Numpy memiliki kemampuan untuk membuat sebuah larik. Larik/array adalah sekumpulan variabel dengan tipe data yang sama.

Numpy harus diinstal terlebih dahulu sebelum digunakan pada sistem ini. Numpy dapat diakses dengan menuliskan "import numpy"



Gambar 3.3 Logo Numpy

## IV. IMPLEMENTASI

Sistem akan membuat sebuah larik dengan menggunakan library Numpy. setelah itu program akan menentukan panjang larik sesuai dengan nilai yang telah ditentukan oleh satu variabel.

Setelah itu sistem akan menginputkan nilai pada setiap elemen larik secara acak menggunakan modul random yang telah diinputkan. Nilai yang akan di inputkan kedalam elemen larik adalah bilangan 1 atau 0. proses penginputan tersebut akan diulang sebanyak panjang larik tersebut.

Setelah semua larik terisi dengan bilangan 1 atau 0, sistem akan mengeluarkan urutan tempat duduk beserta status tempat duduk tersebut. proses pengeluaran itu akan diulang sebanyak panjang larik tersebut.

Setelah itu pengguna diminta untuk memasukkan nomor tempat duduk yang akan diperiksa. Setelah memasukkan nomor tempat duduk tersebut, sistem akan memeriksa apakah tempat duduk tersebut telah dipesan oleh orang lain atau tempat duduk tersebut masih kosong. Jika tempat duduk tersebut masih

kosong, maka pemesanan dapat diproses. Tetapi jika tempat duduk tersebut sudah dipesan oleh orang lain maka program akan mengeluarkan pernyataan.

### A. Source Code Program

```
import random
import numpy as np

x = 5

kursi = np.zeros((x))

for i in range(x):
    kursi[i] = random.randint(0,1)

print("Sistem Pemesanan Kursi Pada Bioskop")
print("")
n = 0
y = int(x)
for i in range(y):
    n = n + 1
    if kursi[i] == 0 :
        print("[", n, "]"',end = " ")
    else :
        print("[", n, "]"',end = " ")
print("")
print("tanda(' ') berarti kursi masih kosong")
print("tanda('1') berarti kursi sudah dipesan")
print("")
inputan = input("Pilih nomor tempat duduk yang akan di pesan : ")

if inputan == "1" :
    if kursi[0] == 1 :
        print("tempat duduk tidak dapat dipesan")
    else :
        print("pemesanan berhasil !!!")
elif inputan == "2" :
    if kursi[1] == 1 :
        print("tempat duduk tidak dapat dipesan")
    else :
        print("pemesanan berhasil !!!")
```

```

elif inputan == "3" :
    if kursi[2] == 1 :
        print("tempat duduk tidak dapat
dipesan")
    else :
        print("pemesanan berhasil !!!")
elif inputan == "4" :
    if kursi[3] == 1 :
        print("tempat duduk tidak dapat
dipesan")
    else :
        print("pemesanan berhasil !!!")
elif inputan == "5" :
    if kursi[4] == 1 :
        print("tempat duduk tidak dapat
dipesan")
    else :
        print("pemesanan berhasil !!!")

```

## B. Pengujian

### a. UJI COBA PERTAMA

Pada uji coba pertama, pengguna mencoba memesan tempat duduk yang masih kosong

```

Sistem Pemesanan Kursi Pada Bioskop

[ 1 ]' [ 2 ]'' [ 3 ]'' [ 4 ]'' [ 5 ]''
tanda(') berarti kursi masih kosong
tanda('') berarti kursi sudah dipesan

Pilih nomor tempat duduk yang akan di pesan
: 1
pemesanan berhasil !!!

```

Pada elemen larik 1, elemen berstatus kosong yang berarti nilai dari elemen larik tersebut adalah 0, sedangkan pada elemen larik 2,3,4,dan 5, elemen berstatus sudah dipesan yang berarti nilai dari elemen larik tersebut adalah 1. Pengguna memasukkan bilangan 1, karena elemen larik berstatus kosong, maka kursi dapat dipesan

### b. UJI COBA KEDUA

Pada uji coba kedua, pengguna mencoba memesan tempat duduk yang masih kosong

```

Sistem Pemesanan Kursi Pada Bioskop

[ 1 ]' [ 2 ]' [ 3 ]' [ 4 ]' [ 5 ]''
tanda(') berarti kursi masih kosong
tanda('') berarti kursi sudah dipesan

Pilih nomor tempat duduk yang akan di pesan
: 3
pemesanan berhasil !!!

```

Pada elemen larik 1,2,3,4, elemen berstatus kosong yang berarti nilai dari elemen larik tersebut adalah 0, sedangkan pada elemen larik 5, elemen berstatus sudah dipesan yang berarti nilai dari elemen larik tersebut adalah 1. Pengguna memasukkan bilangan 3, karena elemen larik berstatus kosong, maka kursi dapat dipesan

### c. UJI COBA KETIGA

Pada uji coba ketiga, pengguna mencoba memesan tempat duduk yang telah dipesan oleh orang lain

```

Sistem Pemesanan Kursi Pada Bioskop

[ 1 ]' [ 2 ]'' [ 3 ]' [ 4 ]' [ 5 ]''
tanda(') berarti kursi masih kosong
tanda('') berarti kursi sudah dipesan

Pilih nomor tempat duduk yang akan di pesan
: 5
tempat duduk tidak dapat dipesan

```

Pada elemen larik 1,3,4, elemen berstatus kosong yang berarti nilai dari elemen larik tersebut adalah 0, sedangkan pada elemen larik 2 dan 5, elemen berstatus sudah dipesan yang berarti nilai dari elemen larik tersebut adalah 1. Pengguna memasukkan bilangan 5, karena elemen larik berstatus sudah dipesan, maka kursi tidak dapat dipesan

#### d. UJI COBA KEEMPAT

Pada uji coba ketiga, pengguna mencoba memesan tempat duduk yang telah dipesan oleh orang lain

```
Sistem Pemesanan Kursi Pada Bioskop

[ 1 ]' [ 2 ]'' [ 3 ]' [ 4 ]' [ 5 ]''
tanda(') berarti kursi masih kosong
tanda('') berarti kursi sudah dipesan

Pilih nomor tempat duduk yang akan di pesan
: 5
tempat duduk tidak dapat dipesan
```

Pada elemen larik 1,3,4, elemen berstatus kosong yang berarti nilai dari elemen larik tersebut adalah 0, sedangkan pada elemen larik 2 dan 5, elemen berstatus sudah dipesan yang berarti nilai dari elemen larik tersebut adalah 1. Pengguna memasukkan bilangan 5, karena elemen larik berstatus sudah dipesan, maka kursi tidak dapat dipesan

VIDEO LINK AT YOUTUBE

-

#### KESIMPULAN

Setelah melakukan pembahasan dan analisa mengenai **Implementasi Algoritma Brute Force Pada Pemilihan Tempat Duduk Bioskop** maka kesimpulan yang dapat diambil adalah :

- Algoritma Brute Force menjadi solusi terbaik untuk menyelesaikan permasalahan yang mengharuskan sistem melakukan pengecekan secara berurutan
- Sistem ini cukup mudah untuk digunakan oleh pengguna dan dapat dikembangkan menjadi sebuah sistem aplikasi karena algoritma yang digunakan sangat sederhana
- sistem ini dapat memudahkan pengguna untuk memesan tempat duduk di bioskop untuk menonton film.

#### REFERENCES

- [1] Munir R, 2004, Algoritma Brute Force, strategi algoritma, Diktat Kuliah, Departemen Teknik Informatika, ITB.
- [2] Hosting J, 2021, Bahasa Pemrograman Python : Pengertian, contoh, dan kegunaan
- [3] Azhari M, 2016, Implementasi Algoritma Brute Force Sebagai Mesin Pencari (Search Engine) Berbasis Web Pada Database
- [4]

#### UCAPAN TERIMA KASIH

Itulah Implementasi dari Algoritma Brute Force pada sistem pemesanan tempat duduk bioskop secara online. Semoga pemaparan yang saya sampaikan cukup jelas untuk dipahami dan mudah untuk dimengerti. Semoga pada makalah kedepannya pemaparan materi dari saya dapat lebih jelas dan lebih rapih lagi. Demikian dari saya, saya ucapkan terima kasih

# 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.

Lampung, 29 Maret 2022



Aqsal Yazid Setiawan  
120140197