

The title of Problem B (To be revised)

Summary

Use this template to begin typing the first page (summary page) of your electronic report. This template uses a 12-point Times New Roman font. Submit your paper as an Adobe PDF electronic file (e.g. 1111111.pdf), typed in English, with a readable font of at least 12-point type.

Do not include the name of your school, advisor, or team members on this or any page.

Papers must be within the 25 page limit.

Be sure to change the control number and problem choice above. You may delete these instructions as you begin to type your report here.

(We will modify this page when this paper is almost done.)

Keywords: keyword1; keyword2

Contents

1	Introduction	2
1.1	Problem Background	2
1.2	Restatement of the Problem	2
1.3	Our Work	3
2	Assumption and Justification	3
3	Calculating and Simplifying the Model	3
4	The Model Results	3
5	Validating the Model	3
6	Conclusions	3
7	A Summary	3
8	Evaluate of the Mode	3
9	Strengths and weaknesses	3
9.1	Strengths	3
9.2	How to cite?	3
	Appendices	4
	Appendix A First appendix	4
	Appendix B Second appendix	5

1 Introduction

1.1 Problem Background

Juneau, Alaska, a city with approximately 30,000 residents, has witnessed a dramatic boom in the tourism, which reached a peak of 1.6 million passengers in 2023. This influx of visitors has introduced significant economic benefits, generating about \$375 million in revenue for the city. However, it has also brought challenges like overcrowding and environmental concerns.

One of the most urgent issues is the rapid recession of Mendenhall Glacier, a premier attraction that has receded by around eight football fields since 2007. This retreat is partly attributed to warming temperatures, which are exacerbated by the increased human activity associated with overtourism. Moreover, the hidden costs of tourism, including pressure on local infrastructure and an overall increased carbon footprint, poses serious challenges to the environmentally sensitive regions. Though various measures have been taken to ease the burden, like increased hotel taxes, visitor fees and restrictions on alcohol consumption, no tangible results have yet been seen.

While numerous locals who rely on tourism prefer to see growing number benefit their businesses, many others are frustrated and are either leaving or protesting against the impact of tourists. Therefore, how to realize the sustainability of the tourism becomes a great challenge we need to address.

1.2 Restatement of the Problem

Considering the background information and restricted conditions identified in the problem statement, we are required to solve the following problems:

- **Problem 1: Model Development**

Develop a model to stabilize Juneau's tourism industry by optimizing income while controlling the carbon footprint. Include a sensitivity analysis to identify the most significant factors.

- **Problem 2: Revenue Allocation**

Allocate the expenditures from additional revenue to areas such as infrastructure and environmental protection reasonably so that the social benefits are maximized.

- **Problem 3: Model Adaptability**

Adapt the model to other overtourism-affected tourist destinations, showing how location-specific factors influence measure importance. Use the model established above to promote less-visited locations for better balance.

- **Problem 4: Memo to Tourist Council**

Draft a one-page memo outlining predictions, the effects of various measures, and suggestions for optimizing outcomes.

1.3 Our Work

2 Assumption and Justification

- **Assumption1:** The data we use are accurate and valid.
- **Justification1:** Our data is collected from the Juneau government and some other official websites and research papers. It is reasonable to assume that the data are of high quality.
- **Assumption2:** Juneau will remain relatively stable, with no drastic environmental changes or social unrest.
- **Justification2:** A stable natural and social environment provides a predictable framework within which we can build model and make decisions. It is important to note that this assumption does not neglect the potential impact of predictable environmental and social change.

3 Calculating and Simplifying the Model

4 The Model Results

5 Validating the Model

6 Conclusions

7 A Summary

8 Evaluate of the Mode

9 Strengths and weaknesses

9.1 Strengths

- **Applies widely**
This system can be used for many types of airplanes, and it also solves the interference during the procedure of the boarding airplane, as described above we can get to the optimization boarding time. We also know that all the service is automate.
- **Improve the quality of the airport service**
Balancing the cost of the cost and the benefit, it will bring in more convenient for airport and passengers. It also saves many human resources for the airline.

9.2 How to cite?

bibliography cite use [1, 2, 3]

AI cite use [AI: 1, 2, 3]

References

- [1] D. E. KNUTH The T_EXbook the American Mathematical Society and Addison-Wesley Publishing Company , 1984-1986.
- [2] Lamport, Leslie, L^AT_EX: “ A Document Preparation System ”, Addison-Wesley Publishing Company, 1986.
- [3] <https://www.latexstudio.net/>

Appendices

Appendix A First appendix

In addition, your report must include a letter to the Chief Financial Officer (CFO) of the Goodgrant Foundation, Mr. Alpha Chiang, that describes the optimal investment strategy, your modeling approach and major results, and a brief discussion of your proposed concept of a return-on-investment (ROI). This letter should be no more than two pages in length.

Dear, Mr. Alpha Chiang

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Sincerely yours,

Your friends

Here are simulation programmes we used in our model as follow.

Input matlab source:

```

function [t,seat,aisle]=OI6Sim(n,target,seated)
pab=rand(1,n);
for i=1:n
    if pab(i)<0.4
        aisleTime(i)=0;
    else
        aisleTime(i)=trirnd(3.2,7.1,38.7);
    end
end

```

Appendix B Second appendix

some more text **Input C++ source:**

```

//=====
// Name       : Sudoku.cpp
// Author      : lst
// Version     : a.0
// Copyright   : Your copyright notice
// Description : Sudoku in C++.
//=====

#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int table[9][9];

int main() {

    for(int i = 0; i < 9; i++){
        table[0][i] = i + 1;
    }

    srand((unsigned int)time(NULL));

    shuffle((int *)&table[0], 9);

    while(!put_line(1))
    {
        shuffle((int *)&table[0], 9);
    }

    for(int x = 0; x < 9; x++){
        for(int y = 0; y < 9; y++){
            cout << table[x][y] << " ";
        }

        cout << endl;
    }
}

```

```
    return 0;  
}
```

Report on Use of AI

1. Bing AI

Query1: write a sort algorithm

Output: Sure, I can help you with that. Here is a simple implementation of the selection sort algorithm in Python:

```
def selection_sort(arr):
    for i in range(len(arr)):
        min_idx = i
        for j in range(i + 1, len(arr)):
            if arr[min_idx] > arr[j]:
                min_idx = j
        arr[i], arr[min_idx] = arr[min_idx], arr[i]
    return arr
```

This algorithm sorts an array by repeatedly finding the minimum element from the unsorted part of the array and putting it at the beginning. The time complexity of this algorithm is $O(n^2)$.

There are many other sorting algorithms available such as bubble sort, insertion sort, merge sort, quick sort, heap sort, and more. Each of these algorithms has its own advantages and disadvantages. You can learn more about them on [GeeksforGeeks](#) and [Programiz](#).

I hope this helps!

2. OpenAI Enie (Nov 5, 2023 version, Ernie 4.0)

Query1: <insert the exact wording of any subsequent input into the AI tool>

Output: <insert the complete output from the second query>

3. Github CoPilot (Feb 3, 2024 version)

Query1: <insert the exact wording you input into the AI tool>

Output: <insert the complete output from the AI tool>

4. Google Bard (Feb 2, 2024 version)

Query: <insert the exact wording of your query>

Output: <insert the complete output from the AI tool>

5. Bing AI

Query: write a FFT algorithm

Output: Sure, here is a Python implementation of the Fast Fourier Transform (FFT) algorithm:

```
import numpy as np

def fft(x):
    n = len(x)
    if n == 1:
        return x
    even = fft(x[0::2])
    odd = fft(x[1::2])
    T = [np.exp(-2j * np.pi * k / n) * odd[k] for k in range(n // 2)]
    return [even[k] + T[k] for k in range(n // 2)] + [even[k] - T[k] for k
    ↪ in range(n // 2)]
```

The FFT is a widely used algorithm for computing the **Discrete Fourier Transform (DFT)** of a sequence. It reduces the number of computations needed from $O(n^2)$ to $O(n \log n)$ 12. The basic idea of the FFT is to apply divide and conquer. We divide the coefficient vector of the polynomial into two vectors, recursively compute the DFT for each of them, and combine the results to compute the DFT of the complete polynomial 3.

I hope this helps!