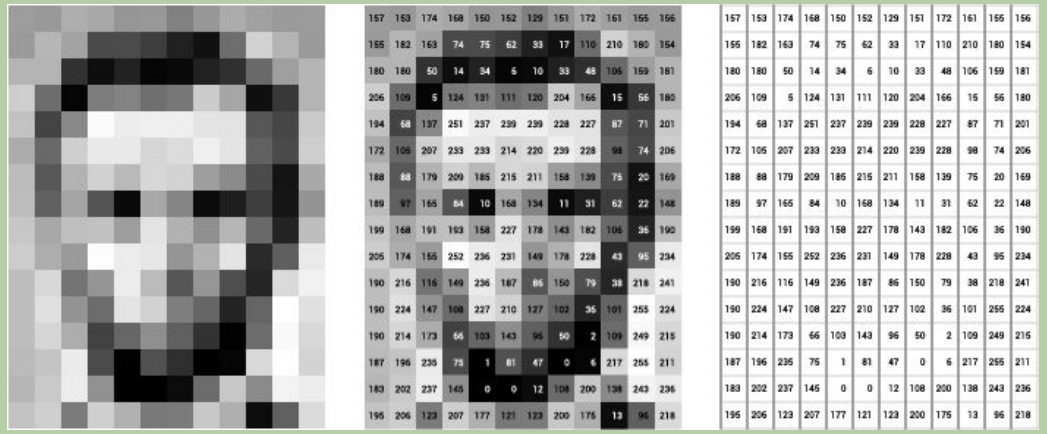
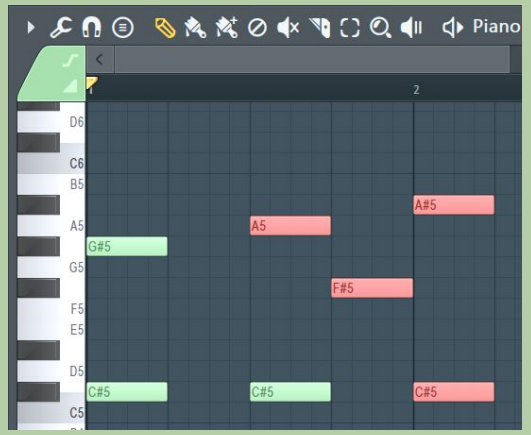





# CS 225 Final Project: PNG-to-Music

Team Patreonson

# Goals

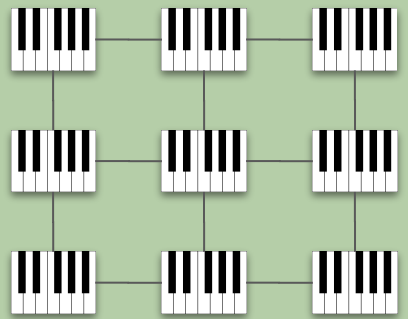
- Graph representation of notes (NoteGraph)
  - Adjacency matrix with edges being weighted distances between notes
- Algorithms used to formulate unique sequences of notes
  - Prim & AStar & DFS
- Turn a picture into music!

# Class Derivations

Key


Note

NoteGraph



# Developments

- NoteGraph
- NoteTable
- A* Search
  - Smart search algorithm
- Prim's Algorithm
  - Greedy algorithm
- DFS Traversal
  - Used to create a sequence of intervals

Original Graph

| | 4G | 5B | 10A | 6F# | 1D | 6C | 8C | 8F | 7F# |
|---|---|---|---|---|---|---|---|---|---|
| 4G | – | 15 | – | 22 | – | – | – | – | – |
| 5B | 15 | – | 57 | – | 56 | – | – | – | – |
| 10A | – | 57 | – | – | – | 56 | – | – | – |
| 6F# | 22 | – | – | – | 63 | – | 17 | – | – |
| 1D | – | 56 | – | 63 | – | 57 | – | 86 | – |
| 6C | – | – | 56 | – | 57 | – | – | – | 17 |
| 8C | – | – | – | 17 | – | – | – | 4 | – |
| 8F | – | – | – | – | 86 | – | 4 | – | 10 |
| 7F# | – | – | – | – | – | 17 | – | 10 | – |

Image 1: Original NoteGraph

```
{{0, Note('G', 4)},
 {1, Note('B', 5)},
 {2, Note('A', 10)},
 {3, Note('f', 6)},
 {4, Note('D', 1)},
 {5, Note('C', 6)},
 {6, Note('C', 8)},
 {7, Note('F', 8)},
 {8, Note('f', 7)}};
```

Image 2: NoteTable

Prim's Algorithm

| | 4G | 5B | 10A | 6F# | 1D | 6C | 8C | 8F | 7F# |
|---|---|---|---|---|---|---|---|---|---|
| 4G | – | 15 | – | 22 | – | – | – | – | – |
| 5B | 15 | – | – | – | 56 | – | – | – | – |
| 10A | – | – | – | – | – | 56 | – | – | – |
| 6F# | 22 | – | – | – | – | – | 17 | – | – |
| 1D | – | 56 | – | – | – | – | – | – | – |
| 6C | – | – | 56 | – | – | – | – | – | 17 |
| 8C | – | – | – | 17 | – | – | – | 4 | – |
| 8F | – | – | – | – | – | – | 4 | – | 10 |
| 7F# | – | – | – | – | – | 17 | – | 10 | – |

Image 3: Prim's NoteGraph

A* Algorithm

| | 4G | 5B | 10A | 6F# | 1D | 6C | 8C | 8F | 7F# |
|---|---|---|---|---|---|---|---|---|---|
| 4G | – | – | – | 22 | – | – | – | – | – |
| 5B | – | – | – | – | – | – | – | – | – |
| 10A | – | – | – | – | – | – | – | – | – |
| 6F# | 22 | – | – | – | – | – | 17 | – | – |
| 1D | – | – | – | – | – | – | – | – | – |
| 6C | – | – | – | – | – | – | – | – | – |
| 8C | – | – | – | 17 | – | – | – | 4 | – |
| 8F | – | – | – | – | – | – | 4 | – | 10 |
| 7F# | – | – | – | – | – | – | – | 10 | – |

Image 4: A* NoteGraph

# Challenges

- **Adjacency matrix**
  - Collapsed image grid from **2D** to **1D**
  - Space complexity of **O((wh)^2)** instead of **O(wh)**
    - A **300x300** image is saved in a **90000x90000** matrix… NOT good
- **Adjacency list**
  - Image can stay in **2D**
  - Space complexity is **O(wh)**
    - Matrix is saved as **300x300**… GOOD

# Conclusions

- It is possible to generate a sequence of notes for a given image.
- What about a **melodic** sequence?...
- Memory efficiency is very important