Politechnika Śląska

Wydział Matematyk Stosowanej

Kierunek Informatyka

Gliwice, 10.02.2022

Programowanie I

**Projekt zaliczeniowy**

**“*Syntezator dźwięków z interfejsem graficznym*"**

**Szymon Hankus gr. lab. 3/5**

**1. Opis projektu.**

Projekt miał na celu stworzenie syntezatora dźwięków z interfejsem graficznym bez użycia zewnętrznych bibliotek do syntezy dźwięków. Synteza odbywa się “manualnie” (bez użycia bibliotek do tego celu) poprzez obliczanie parametrów każdego sample’a. Jedyna biblioteka, którą wykorzystuje projekt - SFML - jest użyta tylko w celu ułatwienia dostępu do urządzeń dźwiękowych na wielu platformach oraz stworzenia rozbudowanego interfejsu graficznego.

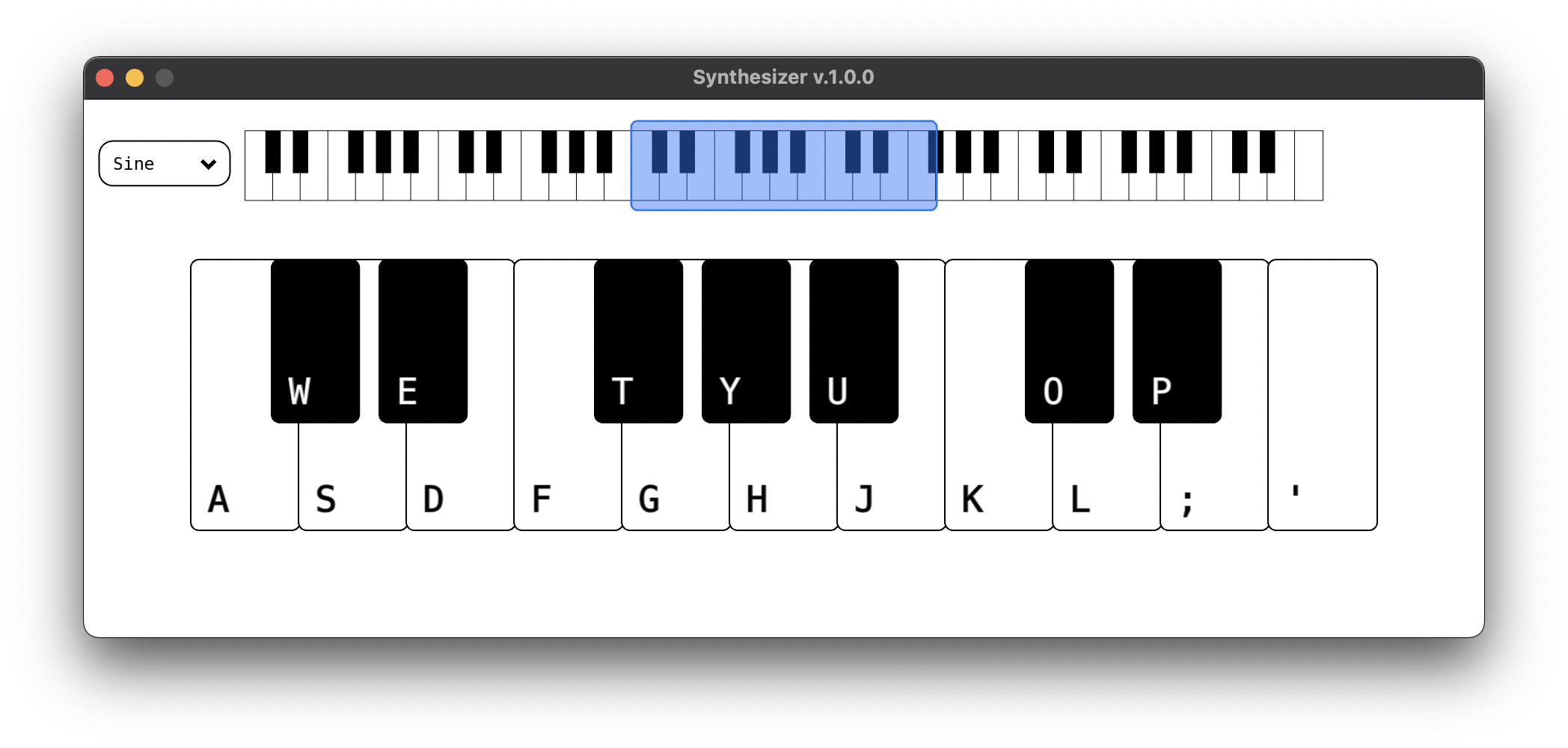
**2. Wymagania**

* Graficzna klawiatura fortepianowa w pięciu oktawach, z możliwością wyboru oktawy na “selektorze oktaw”, czyli elemencie interfejsu graficznego,
* Selektor oktaw - element interfejsu pozwalający na szybki wybór oktawy oraz wskazujący obecną oktawę w sposób przejrzysty,
* Ulepszenie interfejsu poprzez zastosowanie własnoręcznie zaimplementowanych zaokrąglonych prostokątów
* Odzwierciedlenie klawiatury graficznej na fizycznej (np. Dźwięk C to klawisz A na klawiaturze fizycznej, dźwięk D to klawisz S itd.),
* Stosowne animacje podczas naciskania klawiszy zarówno na głównej klawiaturze oraz na selektorze oktaw,
* Synteza dźwięków w pięciu oktawach,
* Trzy typy fali dźwiękowych do wyboru (fala sinusoidalna, fala prostokątna, fala piłokształtna),
* Ładne, dynamiczne menu umożliwiające wybór pomiędzy typami fal
* Możliwość grania melodycznego (pojedyncze dźwięki),
* Możliwość grania harmonicznego (jednocześnie) dowolnej ilości dźwięków

**3. Przebieg realizacji**

Tak, jak wcześniej wspominałem, użyta została tylko biblioteka SFML pozwalająca na łatwy dostęp do systemowego API dźwiękowego oraz graficznego.

Projekt zacząłem od próby syntezy jakiegokolwiek dźwięku. Po nauczeniu się w jaki sposób dźwięk jest przechowywany i odtwarzany na komputerze, udało mi się napisać funkcję zapisującą w tablicy sample prostej fali sinusoidalnej.

Następnie musiałem uogólnić moją metodę dla dowolnej częstotliwości fali. Udało mi się również opracować metody syntezy fal prostokątnych i piłokształtnych.

Stworzyłem prosty interfejs klawiatury z użyciem zwykłych prostokątów oraz dodałem animacje.

Zaimplementowałem klasę zaokrąglonych prostokątów i użyłem ich do zmiany wyglądu klawiatury na bardziej “naturalny”.

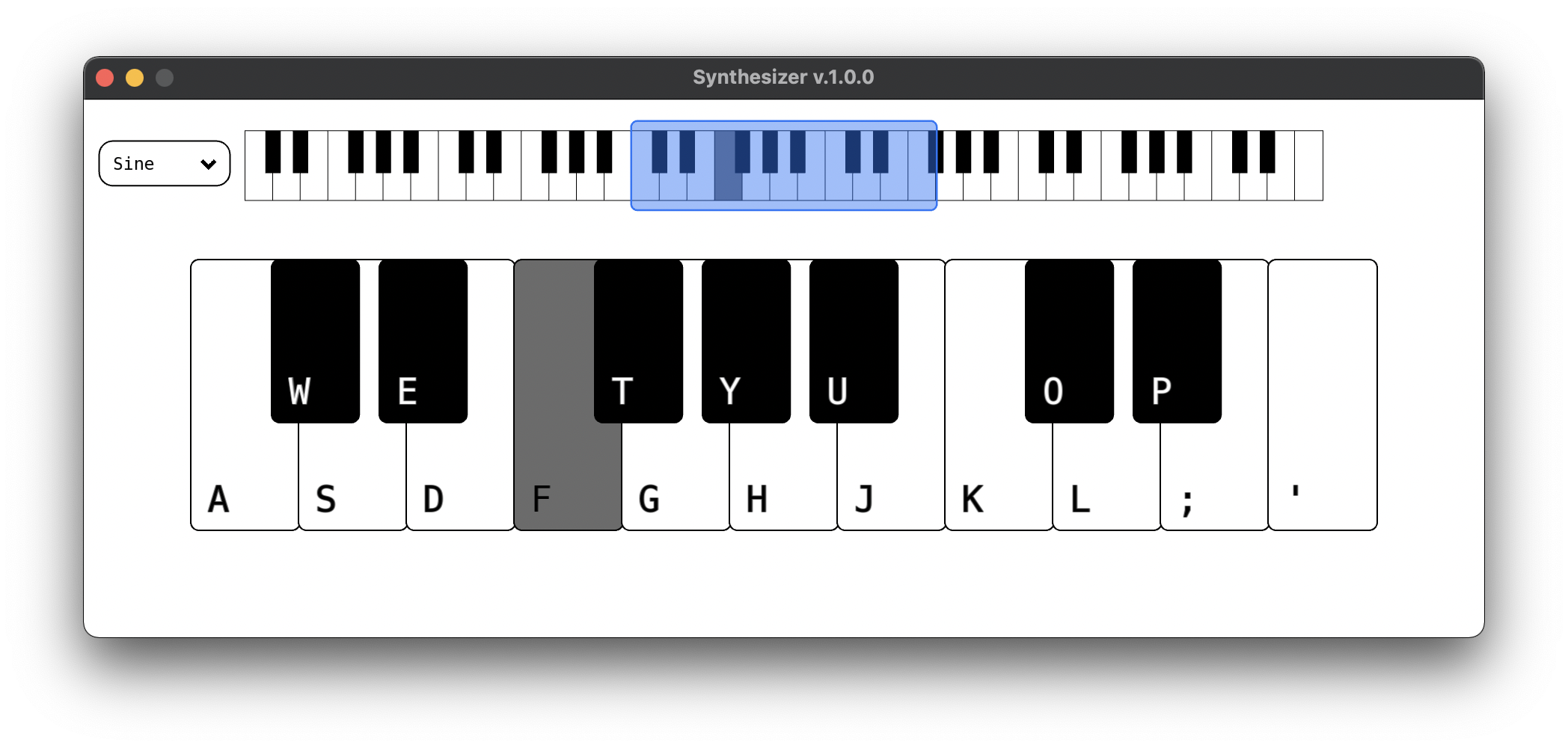
Zaimplementowałem selektor oktaw.

Zaimplementowałem rozwijane menu do wyboru typu fal.

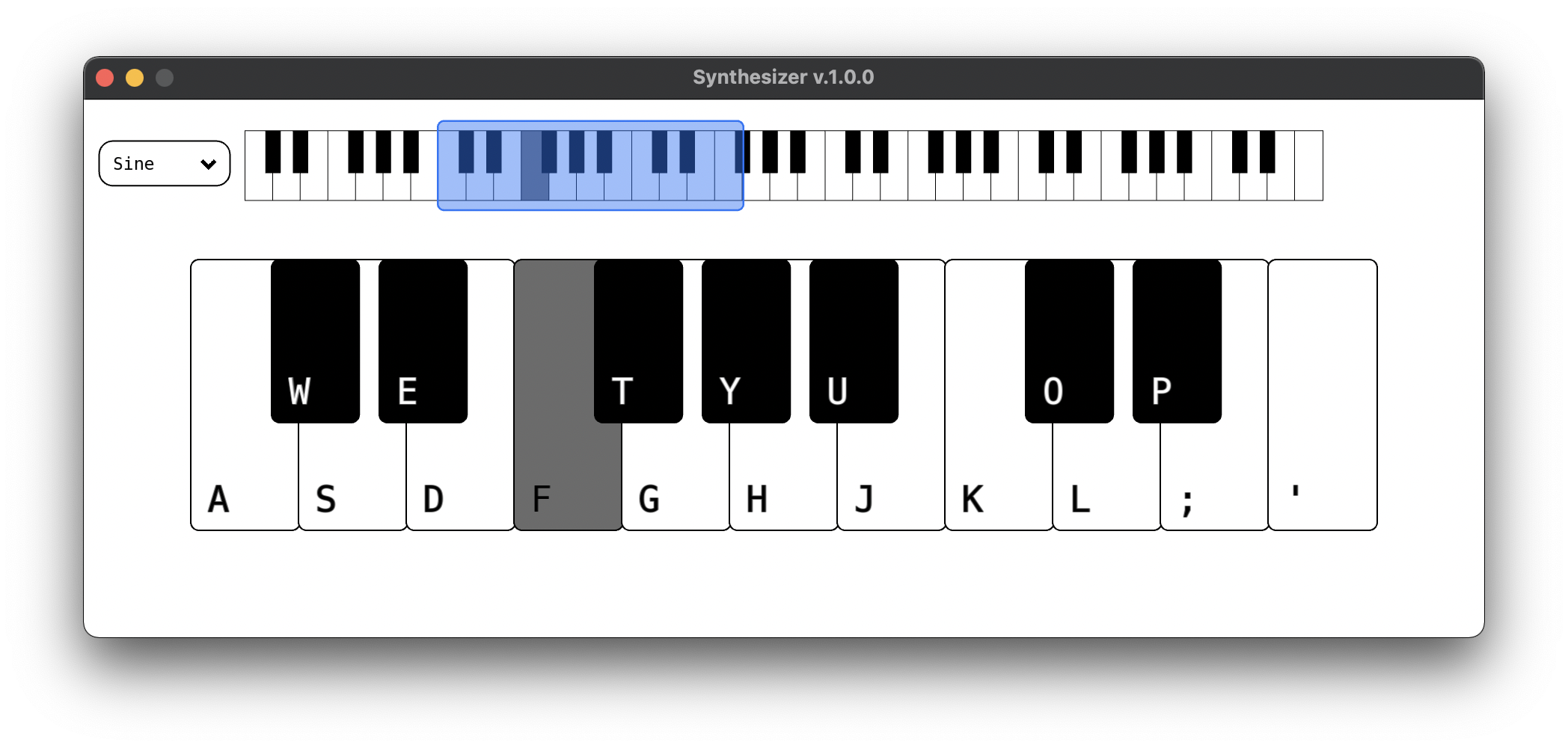
**4. Instrukcja użytkownika**

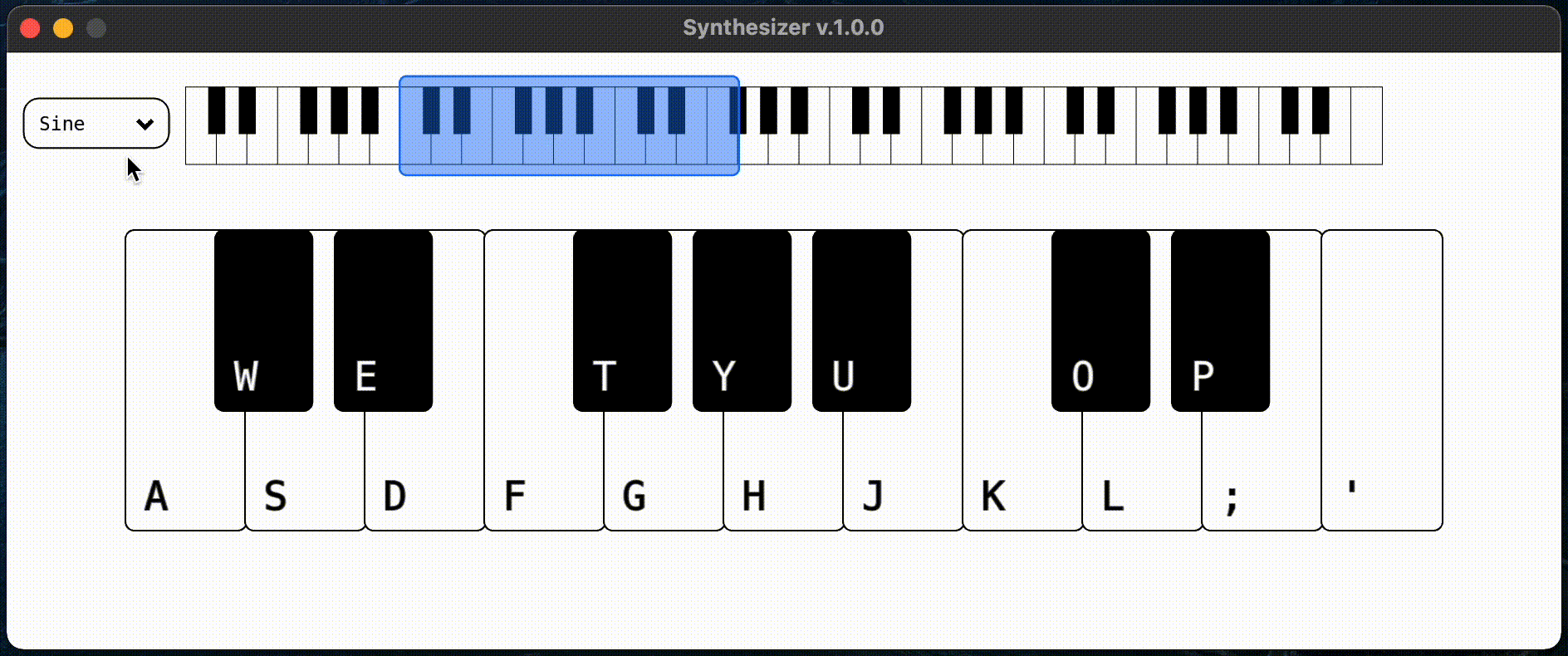
Działanie programu jest bardzo proste i intuicyjne. Interfejs prezentuje się następująco:

Największa klawiatura znajdująca się w samym centrum okna to klawiatura główna. Znajdująca się nad nią długa klawiatura wraz z niebieskim prostokątem to selektor oktaw, wskazujący w której oktawie obecnie znajduje się klawiatura główna. W lewym górnym rogu widnieje rozwijane menu, pozwalające na wybór typu fali.

Aby zagrać pożądany dźwięk wystarczy wcisnąć klawisz, któremu dany dźwięk odpowiada. Po wciśnięciu klawisza (np. F) interfejs zachowa się w następujący sposób:

Zgodnie z oczekiwaniami, podświetlony zostaje odpowiedni klawisz, oraz zagrany zostaje odpowiedni dźwięk.

Aby zmienić oktawę o jedną w dół, należy wcisnąć klawisz Z. Natomiast aby zmienić oktawę o jeden w górę, należy wcisnąć klawisz X. Selektor oktaw przesunie się w odpowiednie miejsce, a częstotliwości dźwięków klawiatury głównej zostaną oczywiście zmienione.

Aby zmienić typ fal, wystarczy kliknąć w menu znajdujące się w lewym górnym rogu, a następnie wybrać żądany typ z listy. Poniżej znajduje się GIF ilustrujący ową funkcjonalność.

**5. Podsumowanie i wnioski.**

Podsumowując, jestem bardzo zadowolony z tego jak udał się projekt. Interfejs jest czysty oraz przejrzysty, a zaimplementowane klasy są łatwo rozbudowywalne i przerabialne. W przyszłości z ich użyciem można by na przykład zaimplementować odtwarzarkę plików MIDI, bądź też nagrywarkę.