

## Flight Recorder

### 1. Ce

Celem projektu jest zrobienie programu łączącego funkcjonalność czarnej skrzynki i przyrządów pomiarowych znajdujących się przykładowo w samolotach pasażerskich.

### 2. Dane pokładowe obiektu latającego

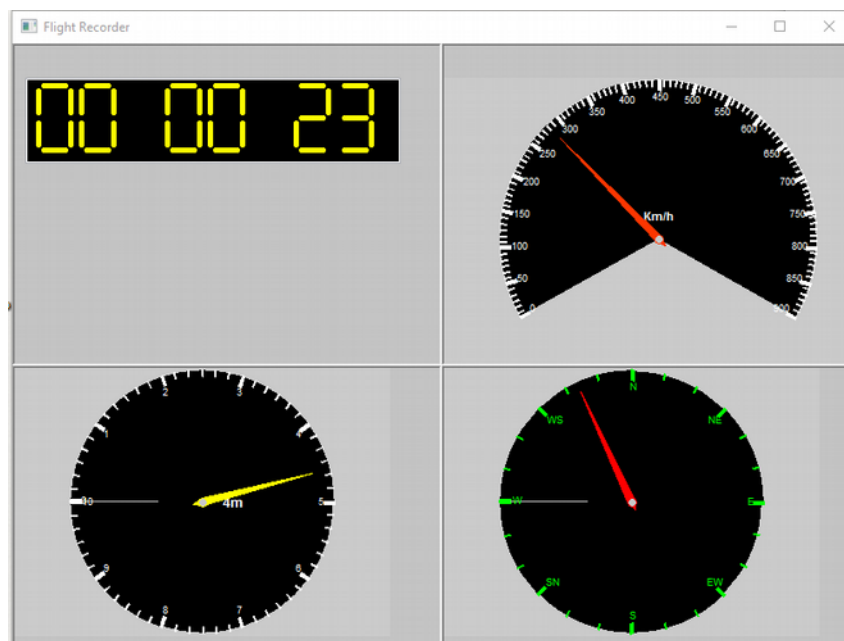
Pierwszym problemem, z którym musiałem sobie poradzić było to skąd pozyskać dane obiektu latającego, które można przekierować do programu. W pierwszej kolejności postanowiłem znaleźć autentyczne dane lotów, nie jest to łatwe zadanie jednak wykonalne i można je pobrać ze strony : <https://c3.nasa.gov/dashlink/projects/85/resources/?type=ds> . Posiadają rozszerzenie .mat, co wskazuje na dane programu matlab. Dane te odczytałem przy użyciu biblioteki SciPy, jednak okazało się, że ilość danych i mój brak wiedzy w zakresie danych przekazywanych przez samolot nie pozwoliły mi na ich analizę. Ponieważ uczenie się tego było by zbyt czasochłonne postanowiłem pobrać dane z już istniejącego symulatora. Okazało się, że jedynym symulatorem, posiadającym API do pobierania danych jest [Microsoft Flight Simulator](#), a API jest tylko dla platformy .NET.

Z powodu powyższych niedogodności musiałem stworzyć własny „symulator” samolotu. Symuluje on tylko najprostsze dane, głównie w sposób liniowy i niezależny od siebie, są to m.in. prędkość, wysokość, kąt wznoszenia, czy kierunek. Prędkość rośnie wykładniczo do wartości ok. 120 km/h, następnie maleje przyspieszenie i wzrost jest eksponencjalny. Wysokość rośnie liniowo, kierunek zmienia się w zależności od wylosowanych liczb pseudolosowych.

Prędkość maksymalna została tak dobrana aby nie odbiegała rzeczywistych osiąarów samolotów i wynosi 900 km/h, szybkość wznoszenia jest liniowa i wynosi 60m/min, co odpowiada rzeczywistej średniej prędkości wspinaczki samolotu.

### 3. Prezentacja danych

Dane są prezentowane przy pomocy biblioteki wxPython. Pozwala ona w prosty sposób stworzyć liczniki (poniższy screen) i dowolnie je modyfikować w zależności od potrzeb. W równie prosty sposób można stworzyć zegar LED-owy. Pierwszy z liczników to prędkościomierz, drugi wysokościomierz a trzeci to kompas.



Po odpowiednim skonfigurowaniu liczników, stworzyłem funkcje zmieniające wartości prędkości, wysokości i kierunku, tak aby GUI działało możliwie jak najpłynniej. Warto również wspomnieć, że wielowątkowość przy wykorzystaniu wxPythona nie jest rzeczą trywialną i osiąga się ją dużo prościej w połączeniu z biblioteką qt.

#### 4. Zapisywanie danych

Dane są zapisywane w bazie postgresql. Baza na potrzeby prezentacji programu została umieszczona na zewnętrznym serwerze. Tabela w której zapisywane są dane nazywa się flight, a kolejne kolumny nazwane są : speed, alt, climb, direction, banking, fuel . Danych tych jest mało ponieważ mieliśmy stworzyć „initial project”, czyli o ile dobrze zrozumiałem przedstawienie pomysłu na działanie całości programu. W obecnej wersji zapisywane są tylko prędkość, wysokość, kąt wznoszenia i kierunek.

Do połączenia z bazą danych wykorzystałem bibliotekę psycopg2 i stworzyłem własną funkcję zapisującą wiersze w tabeli flight.