

# Dokumentacja projektu - system zapisów w II w semestrze letnim

Mateusz Gieroba (322072)

14 lutego 2021

## 1 Obsługa

### 1.1 Kompilacja i uruchamianie

Kompilacja odbywa się poprzez wykonanie w terminalu polecenia

```
make main
```

Program uruchamia się poleceniem

```
./main
```

### 1.2 Wprowadzenie

System obsługuje cały okres semestru letniego, tj. od 24 lutego 2021 do 30 września 2021. Z tego powodu pomija się wypisywanie roku w jakiegokolwiek interakcji z systemem. Pozwala on na utrzymywanie bazy danych urzędzeń, tworzenie i usuwanie zdarzeń ich wypożyczania oraz zadawanie złożonych zapytań dotyczących samych urzędzeń, a także ich wypożyczeń i dostępności. Interfejs opiera się całkowicie na terminalu systemowym, bazy utrzymuje w plikach tekstowych, przyjmuje polecenia z terminala oraz wypisuje je w zależności od preferencji użytkownika w terminalu bądź do pliku. Menu jest dokładnie opisane, więc nie powinny zaistnieć problemy z obsługą systemu.

### 1.3 Nawigacja w menu głównym

Wywoływanie odpowiednich funkcji programu odbywa się przy pomocy jednoliterowych komend wpisywanych w terminalu (po wprowadzeniu *I* możemy przeczytać je na ekranie). Jeżeli użytkownik przypadkowo wciśnie *ENTER* nie wprowadziwszy uprzednio jednoliterowego polecenia powinien ponownie wcisnąć *ENTER*. Czasami funkcje zadają pytania o chęć zapisu do pliku bądź potwierdzenie działania - odpowiedź to **Y** - TAK bądź **N** - NIE. Funkcje udostępniane w interfejsie i ich komendy to:

- **A** - dodanie urzędzenia do bazy urzędzeń
- **B** - wypożyczenie urzędzenia
- **D** - usunięcie urzędzenia z bazy dostępnych do wypożyczenia urzędzeń
- **I** - wypisanie komend menu głównego
- **L** - wypisanie listy urzędzeń (danego typu bądź wszystkich aktywnych)
- **M** - modyfikacja rekordu urzędzenia w bazie
- **N** - wypisanie danych konkretnego wypożyczenia
- **R** - skasowanie wypożyczenia i zwolnienie urzędzenia w zajmowanym przez nie terminie
- **Q** - zapisanie baz i zamknięcie interfejsu
- **W** - wypisanie urzędzeń (wszystkich bądź danego typu) dostępnych w zadanym terminie
- **X** - wypisanie tygodniowego planu wypożyczeń
- **Y** - wypisanie wszystkich rezerwacji podpisanych przez wybraną osobę

**UWAGA** Program powinien być zamykany tylko przy użyciu polecenia '**Q**'. W razie wystąpienia konieczności siłowego zamknięcia programu należy najpierw wykonać kopie zapasowe plików *wypozyczenia.txt* oraz *urzedzenia.txt*.

## 2 Moduły

### 2.1 Urządzenia

Moduł ten utrzymuje bazę danych urządzeń o parametrach: ID, typ, nazwa, opis, informacja czy urządzenie nie zostało skasowane oraz pomocnicza tabela do zaznaczania wynajęcia w danym terminie. Udostępnia on funkcje:

- *wczytajLinie* - funkcja pomocnicza do wczytywania tekstu o ograniczonej długości
- *wczytajUrządzenia* - funkcja wczytująca dane z pliku tekstowego do odpowiednich struktur
- *zapiszUrządzenia* - funkcja w menu głównym zapisująca dane do pliku tekstowego
- *rozszerzUrządzenia* - funkcja pomocnicza powiększająca rozmiar tabeli struktur urządzeń
- *dodajUrządzenie* - funkcja dodająca nowe urządzenie do bazy
- *wypiszUrządzenia* - funkcja wypisująca wszystkie aktywne urządzenia w bazie, można sparametryzować po typie urządzenia oraz wypisać do pliku
- *usunUrządzenie* - funkcja usuwająca urządzenie z bazy (dezaktywująca je dla innych funkcji)
- *modyfikujUrządzenie* - funkcja modyfikująca dane urządzenia już wprowadzonego do bazy

### 2.2 Wypożyczenia

Funkcje zdefiniowane w tym module utrzymują bazę danych rezerwacji (zwanymi też wypożyczeniami) oraz przetwarzają je do postaci, która jest wygodna do odpowiadania na zapytania. Struktura wypożyczenia składa się z jego ID, ID wypożyczanego urządzenia (wiąże bazę wypożyczeń z bazą urządzeń), liczb oznaczających początek i koniec wypożyczenia (gdzie 0 to 24/02 00:00), informację o tym, czy jest to cykliczne wypożyczenie (tj. czy co tydzień w podanych godzinach urządzenie będzie wypożyczane), imię i nazwisko osoby wypożyczającej oraz informację, czy ta rezerwacja nie została skasowana. Funkcje w tym module to:

- *dataNaLiczbe* - konwertuje datę w postaci dnia, miesiąca i godziny na liczbę, którą używa się do kodowania terminów wypożyczeń
- *liczbaNaDate* - funkcja odwrotna do powyższej
- *wczytajWypożyczenia* - wczytuje bazę wypożyczeń z pliku tekstowego i zapisuje do odpowiednich struktur
- *czyDostepneWDanymOkresie* - sprawdza dostępność danego urządzenia w zadanym przedziale czasu
- *zajmijOkresDostepnosc* - funkcja wiążąca jakiś okres dostępności urządzenia z konkretnym wypożyczeniem
- *dodajWypożyczenie* - pozwala na rezerwację urządzenia
- *usunWypożyczenie* - usuwa rezerwację urządzenia
- *wypiszWypożyczenia* - wypisuje dane wszystkich bądź konkretnego wypożyczenia
- *zapiszWypożyczenia* - zapisuje dane o wypożyczeniach do pliku tekstowego
- *rozszerzWypożyczenia* - funkcja pomocnicza podobna do tej z modułu urządzeń
- *usunRezerwacjeSkasowanegoUrządzenia* - dezaktywuje rezerwacje dokonane na usuniętym z systemu urządzeniu

## 2.3 Zapytania

- *wyznaczPoczątekTygodnia* - funkcja pomocnicza do wypisywania tygodniowego planu wypożyczeń, podaje kod godziny 00:00 w poniedziałek w tygodniu, w którym zawiera się podana data
- *dostępnośćUrządzeń* - wypisuje wszystkie urządzenia (bądź tylko danego typu) dostępne w zadanym przedziale czasu
- *drukujPlanWypożyczeń* - wypisuje tygodniowy plan wszystkich wypożyczeń poprzez wypisanie wszystkich urządzeń wypożyczonych w danej godzinie
- *wypożyczeniaOsoby* - wypisuje wszystkie aktywne rezerwacje dokonane na dane zadanej osoby

## 3 Podsumowanie realizacji założeń

Patrząc na składany wcześniej opis projektu stwierdzam, że udało mi się zrealizować wszystkie stawiane tam wymagania. System miał być jednak prosty, toteż nie stosowałem złożonych struktur danych do przeszukiwania wypożyczeń, uznając, że będą to małe bazy i bardziej zależeć użytkownikom będzie na zajmowaniu małej ilości zasobów przez program. Dodatkowo, rozszerzyłem projekt o pewne rodzaje zapytań, których w pierwotnym planie nie było. Starałem się też zaimplementować jak najwięcej elementów poznanych podczas zajęć - m.in. wskaźniki i dynamicznie alokowana pamięć, zapis i odczyt z plików, typy wyliczeniowe czy też struktury. Tworzenie tego projektu było bardzo uczącym doświadczeniem (szczególnie w kwestii wczytywania napisów, które nie jest takie bez troskie jak mogłoby się wydawać) i myślę, że w przyszłości postaram się rozszerzyć tę pracę o interfejs graficzny i możliwości większego modelowania zapytań przez użytkownika.