

Zad. 7.1

Napisz program pozwalający na pracę z bazą danych asortymentu sklepu. Program powinien pozwalać na:

1. Wczytanie listy produktów z pliku
Plik typu csv o następującej budowie:
NumerProduktu, NazwaProduktum, Wytwórca
2. Przeglądanie zawartości listy produktów
3. Wyszukiwanie produktu z listy na podstawie dowolnego argumentu (numer produktu, nazwa produktu albo wytwórca)
4. Sortowanie listy wyszukiwania na podstawie dowolnego argumentu (numer produktu, nazwa produktu albo wytwórca)
5. Dodawanie produktu do bazy
6. Usuwanie produktu z bazy
7. Zapis zmodyfikowanej listy produktów do pliku przy zakończeniu działania programu

Program powinien zostać napisany w architekturze maszyny stanów.

Zad. 7.2

Zbuduj prosty program „Keypad”.

Wciśnięte liczby powinny wyświetlać się w kontrolce *Pressed numbers*. Po wciśnięciu *Reset record*, zawartość kontrolki *Pressed numbers* powinna zostać wyczyszczona. Po wciśnięciu przycisku *Stop* program powinien zatrzymać się.

7	8	9
4	5	6
1	2	3
	0	

Pressed numbers

8453

Reset record

Stop

Zad. 7.3

Napisz program „Automatic beer pouring machine”.

Program powinien rozpoczynać się z pełnym kuflem piwa (kontrolka typu *slider* – zakres max. 500 ml). Użytkownik wciskając przycisk *Drink beer* powinien móc upijać piwo z kufła. Wielkość łyku powinna być ustawiana z kontrolki *Draught level [ml]*.

Jeżeli poziom piwa w kufłu spadnie poniżej 50 ml, automat sam powinien uzupełnić kufel do maksimum.

Zakończenie programu jest możliwe po wciśnięciu przycisku *No more drinking*, jednak po min. 2 krotnym opróżnieniu kufła.



Zad. 7.4

Zbuduj program „Portable water filter”.

W momencie wciśnięcia przez użytkownika przycisku *Run cleaning*, program powinien pobrać próbkę wody i sprawdzić czy jej zanieczyszczenie przekracza 0,2%. Jeżeli stan zanieczyszczenia został uznany za zbyt duży, program powinien uruchomić operację oczyszczania wody, po czym ponownie sprawdzić stopień zanieczyszczenia. Jeżeli zanieczyszczenie jest dalej zbyt duże, system ponownie powinien spróbować oczyścić wodę, sprawdzić stopień zanieczyszczenia itd.

Oczyszczanie i sprawdzanie powinny być wykonywane w osobnych stanach. W stanie oczyszczania nie musisz implementować żadnej szczególnej funkcjonalności. W stanie sprawdzania, stopień zanieczyszczenia zasymuluj funkcją *random <0,1>*.

Maksymalna ilość prób oczyszczenia wody to 3 razy. Po trzech próbach, program powinien poinformować o wystąpieniu problemu w kontrolce *Message* i *Result*. Jeżeli nie ma problemu, kontrolka *Message* nie powinna wyrzucać żadnego komunikatu, a kontrolka *Result* powinna się świecić na zielono.

The screenshot shows a graphical user interface for a 'Portable water filter' simulation. At the top, there are two buttons: 'Run cleaning' and 'Stop'. Below these buttons is an 'Info' panel. Inside the 'Info' panel, there are three input fields: 'Attempts to clean' with the value '3', 'Last contamination [%]' with the value '0.3', and 'Message' with the text 'It's too contaminated!'. At the bottom of the 'Info' panel, there is a red circular indicator labeled 'Result'.