Politechnika Śląska
Wydział Matematyki Stosowanej
Kierunek Informatyka

Gliwice, 29.01.2022

Programowanie I

Projekt zaliczeniowy

"*Co zjeść?*"

Aleksandra Kacprzak, gr. 1/1

1. Opis projektu.

Aplikacja „Co zjeść?” ma na celu pomóc w codziennym wyborze posiłków. Poprzez podanie przez użytkownika składników, jakie ma w domu, program wyszukuje w bazie posiłków, które są możliwe do stworzenia. Jednocześnie, użytkownik może wyświetlić dokładny przepis wykonania danej potrawy. Dzięki możliwości dodawania własnych przepisów, usuwania ich oraz wyświetlania pełnej bazy użytkownik może mieć wszelkie swoje ulubione przepisy w jednym miejscu.

2. Wymagania

Program zawiera następujące funkcjonalności:

- Możliwość wyświetlania przepisów wpisanych do bazy.
- Dodawanie własnych przepisów.
- Usuwanie istniejących przepisów.
- Dynamiczne przygotowywanie listy składników możliwych do wybrania przez użytkownika na podstawie przepisów wprowadzonych do programu. Dzięki temu można mieć pewność, że dla każdego składnika wyświetlanego na liście można znaleźć przynajmniej jedną zawierającą go potrawę.
- Możliwość wybierania przez użytkownika składników, które posiada i w następstwie generowanie listy potraw, które są możliwe do wykonania z nich.
- Wyświetlenie przepisu dla wybranej potrawy z wygenerowanej listy możliwych posiłków.

3. Przebieg realizacji

Projekt „Co zjeść?” składa się z następujących plików:

- main.cpp – główny plik .cpp zawierający funkcję main(), w której wywołana jest funkcja początkowa programu, czyli menu(). W pliku tym załączony jest również plik nagłówkowy functions.hpp.
- functions.hpp – plik nagłówkowy zawierający deklaracje funkcji wykorzystywanych w programie,klasy recipes oraz biblioteki iostream, fstream i vector. Na początku pliku dołączone są również wszelkie biblioteki potrzebne do działania programu.

Opis bibliotek użytych w projekcie zawarty jest w poniższej tabeli.

Nazwa biblioteki	Opis biblioteki	Co zostało użyte w programie?
iostream	Standardowa biblioteka wyjścia i wejścia.	Strumienie wejścia i wyjścia: cout, cin, i powiązane z nimi funkcje (przykładowo czyszczące strumienie).
fstream	Biblioteka pozwalająca na odczyt plików i zapis do nich.	Zmienne fstream oraz ofstream, otwieranie plików oraz zapis do nich a także zamykanie plików.
regex	Biblioteka służąca do obsługi wyrażeń regularnych, za pomocą których możliwe jest walidowanie łańcuchów znakowych.	Zmienna regex, do której wpisane zostało wyrażenie regularne, zmienna smatch, która jest niezbędna do funkcji regex_match, której celem jest stwierdzenie czy dany łańcuch znakowy spełnia warunki wyrażenia regularnego.

- recipesBase.txt – plik tekstowy pełniący funkcję bazy. Zapisane są w nim: nazwa potrawy, składniki potrzebne do wykonania jej oraz przepis. Każdy kolejny „rekord” jest w kolejnej linii. Schemat każdej linii prezentuje się następująco:

1. name='nazwaPotrawy' ingredients=[skladnik1,skladnik2] recipe='przepis’

- functions.cpp – plik zawierający definicje wszystkich funkcji wykorzystywanych w programie. Krótki opis wszystkich funkcji jest w tabeli poniżej.

Nazwa funkcji	Opis funkcji
fstream openFile(string file)	Otwiera plik o podanej w argumencie nazwie o typie string. W ścieżce pliku wykonywalnego sprawdza istnienie pliku, a jeżeli nie potrafi go odnaleźć, proponuje utworzenie nowego pliku. Jeżeli użytkownik nie wyrazi chęci utworzenia go, program zostaje zakończony. Z kolei, jeżeli plik jest poprawnie otwarty lub utworzony, funkcja ta zwraca zmienną typu fstream.
vector<string> separateBy(string str, char by, bool deleteSpaces)	Rozdziela podany łańcuch tekstowy, który jest podzielony określonym znakiem. Trzeci argument określa, czy w danym tekście usuwamy spacje czy zamierzamy je pozostawić. Zwraca ona wektor, w którym zawarte są podzielone fragmenty (zmienne typu string).
vector<recipes> analyseRecipeFile(fstream &file)	Analizuje plik z przepisami i porządkuje powiązane z nimi informacje do obiektów klasy

	recipes. Obiekty te zawierają nazwę, składniki oraz przepis. Przewiduje ona możliwość pustego pliku i w takiej sytuacji zwraca pusty wektor. W innym przypadku analizuje plik i linijka po linijce tworzy obiekt, które następnie zapisuje w wektorze. Zwraca ona wektor z kolejnymi obiektami klasy recipes.
void addRecipe()	Pozwala ona dodawać użytkownikowi własne przepisy. Pyta użytkownika kolejno o nazwę przepisu, składniki oraz przepis, które następnie zapisuje w odpowiedni sposób w pliku z przepisami.
void deleteRecipe()	Pozwala ona usuwać użytkownikowi przepisy. Wyświetla listę zawierającą nazwy przepisów i pyta się użytkownika, który chce usunąć. Następnie szuka posiłku o danej nazwie i jeżeli go odnajdzie – usuwa go, w przeciwnym przypadku zgłasza komunikat „Nie znaleziono takiej potrawy”. Funkcja ta działa na zasadzie tworzenia nowego pliku, w którym zapisane są wszystkie linie z pliku pierwotnego, poza tą zawierającą usuwany przepis. Pierwotny plik jest usuwany, a następnie zmieniana jest nazwa nowego pliku na „recipesBase.txt”.
vector <string> searchIngredients()	Tworzy listę składników, które są używane w przepisach. Zwraca wektor zawierający zmienne typu string z kolejnymi składnikami,
vector <string> chooseIngredients(vector <string> Ingredients)	Pyta użytkownika o numery składników, które ten posiada u siebie w domu. Następnie za pomocą wyrażeń regularnych sprawdza poprawność podanego przez użytkownika ciągu znaków. Gdy wejście jest poprawne funkcja zwraca wektor zawierający zmienne string, w których zapisane są wybrane składniki, natomiast w przypadku odwrotnym program wraca do menu głównego.
vector <recipes> searchRecipes(vector <string> chosenIngredients)	Szuka przepisów, które są możliwe do zrobienia z podanych przez użytkownika składników. Jeżeli nie znajdzie żadnego – wyświetla komunikat z taką informacją, natomiast w innym przypadku wypisuje możliwe posiłki i pyta się użytkownika, czy chciałby zobaczyć sposób wykonania któregoś z nich. W przypadku niepoprawnego wejścia program wraca do menu głównego, natomiast, gdy jest poprawne – wyświetla przepis i zwraca wektor z obiektami zawierającymi możliwe przepisy.
bool isNumber(const string& str)	Sprawdza, czy dany string zawiera liczbę całkowitą.

void show()	Wypisuje wszystkie przepisy wpisane do bazy.
void searchingRecipes()	Zbiera funkcje służące do wygenerowania możliwych posiłków z podanych składników. Zawiera funkcje: searchIngredients, chooseIngredients, searchRecipes.
void menu()	Wyświetla menu i pozwala użytkownikowi wybrać dane działanie.
void showBottomMenu()	Wyświetla „małe menu”, które pozwala użytkownikowi podjąć decyzję czy chce wrócić do menu głównego, czy zakończyć program.

Na początku pliku functions.cpp użyta jest instrukcja using namespace std, która określa używaną w pliku przestrzeń nazw oraz dołączony jest plik nagłówkowy functions.hpp. Wprowadzona jest również globalna zmienna typu bool o nazwie endProgram, której celem jest determinowanie dalszego działania programu lub jego zatrzymania.

4. Instrukcja użytkownika

Po włączeniu programu widoczne jest następujące menu:

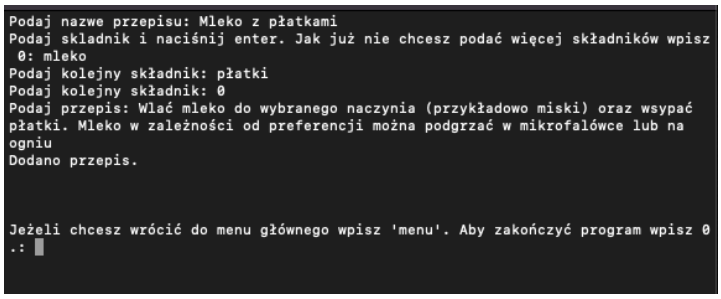
```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@ Witaj w aplikacji "Co zjeść?" Została ona stworzona,    @
@ ułatwić Ci wybór potrawy na posiłek. Możesz do niej     @
@ również dodawać własne przepisy, usuwać bieżące. Do     @
@ obsługi będzie Ci służyć klawiatura.                     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@                         MENU:                            @
@   1. Wyświetl wszystkie przepisy.                        @
@   2. Dodaj nowy przepis.                                 @
@   3. Usuń przepis.                                       @
@   4. Wygeneruj możliwe posiłki z podanych składników.    @
@   5. Wyjdź                                               @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@   Co wybierasz?:
```

Numery kolejnych pozycji w menu wskazują co należy wpisać, aby przejść do danego segmentu. Przykładowo po wpisaniu „1”, wyświetlone zostaną wszystkie posiłki wpisane do bazy.

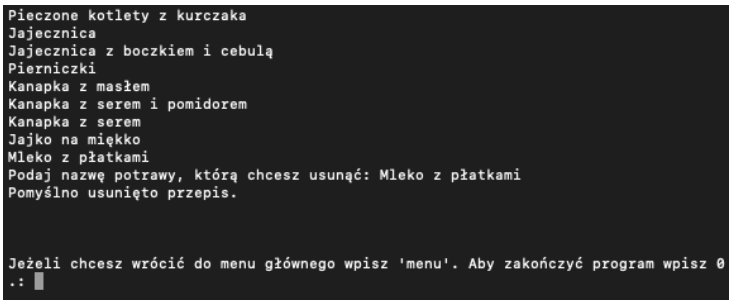
```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Pieczone kotlety z kurczaka
Składniki: pierś z kurczaka, olej rzepakowy, jajka, mleko, bułka tarta, parmezan
Przepis: Pierś  z kurczaka pokrój na mniejsze kawałki o grubości max. 3 cm. Jeśl
i używasz polędwiczek to nic z nimi nie trzeba robić. Mięso natrzyj przyprawami
i olejem – odstaw na co najmniej 30 minut. Jajka rozmieszaj z mlekiem. Bułkę tar
tą zmiksuj z parmezanem. Możesz też parmezan zetrzeć na tarce do bułki tartej.
Kawałki mięsa obtaczaj najpierw w jajku, a następnie w panierce z bułki tartej.
Napanierowane mięso układaj na blaszce wyłożonej papierem do pieczenia i skrop p
o wierzchu olejem rzepakowym. Piecz w piekarniku rozgrzanym do 200 stopni C prze
z ok. 30 minut. W połowie pieczenia obróć na drugą stronę.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Jajecznica
Składniki: jajka
Przepis: Jajka rozbijamy za pomocą trzepaczki. Wlewamy jajka na patelnię i miesz
amy.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Jajecznica z boczkiem i cebulą
Składniki: jajka, boczek
Przepis: Pokrojoną cebulę wrzucamy na patelnię i czekamy, aż się zeszkli. Jajka
rozbijamy za pomocą trzepaczki. Do podsmażonej cebuli dodajemy boczek i lekko po
dsmażamy. Wlewamy jajka na patelnię i cały czas mieszamy.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Pierniczki
Składniki: mąka pszenna, jajka, cukier puder, masło, miód, kakao, soda oczyszczo
na
Przepis: Z mąki pszennej, jajek, cukru pudru, masła, miodu, kakao i sody oczyszc
zonej zagniatamy ciasto. Będzie się lekko lepić ale nie dodajemy mąki. Następnie
 lekko podsypujemy stolnicę mąką i rozwałkowujemy na grubość ok 4-5 mm. Następni
e wycinamy dowolne kształty i pieczemy w piekarniku rozgrzanym do 180 stopni w z
ależności od ich wielkości 8-10 minut.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Kanapka z masłem
Składniki: chleb, masło
Przepis: Posmaruj chleb masłem
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Kanapka z serem i pomidorem
Składniki: chleb, masło, ser żółty, pomidor
Przepis: Posmaruj chleb masłem i połóż na nim żółty ser. Następnie pokrój pomido
ra i nałóż na chleb
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Kanapka z serem
Składniki: chleb, masło, ser żółty
Przepis: Weź kromkę chleba i posmaruj ją masłem. Następnie połóż na niej ser żół
ty.
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Nazwa przepisu: Jajko na miękko
Składniki: jajka
Przepis: Umieść jajko w rondelku, zalej je zimną wodą i dodaj odrobinę soli. Gdy
 woda zacznie wrzeć, rozpocznij odmierzanie czasu. Wystarczą 3-4 minuty, aby uzy
skać idealne jajko na miękko.



Jeżeli chcesz wrócić do menu głównego wpisz 'menu'. Aby zakończyć program wpisz
0.:
```

Na samym dole wyświetla się „menu”, które pozwala nam zakończyć program (po wpisaniu 0) lub przejść do menu (po wpisaniu menu). W przypadku, jeżeli zostanie wpisana niepoprawna wartość, program wróci do menu.

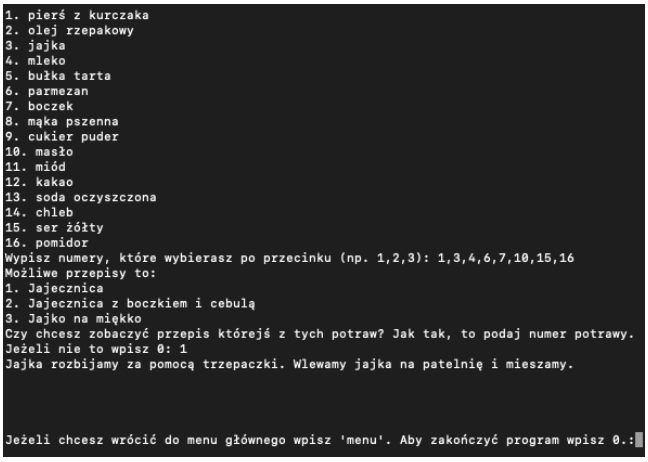
Aby dodać nowy przepis, należy w menu wpisać pozycję „2”. Wówczas kreator pyta się o nazwę posiłku, składniki (aby zakończyć wpisanie ich należy wpisać 0) oraz sposób przyrządzenia. Przykład poniżej:



Aby usunąć przepis należy użyć opcji numer 3 w menu. Wówczas program wyświetla listę przepisów i aby któryś z nich usunąć należy wpisać jego nazwę. Jeżeli dana nazwa nie zostanie odnaleziona, program wyświetli komunikat informujący o nieznalezieniu podanego posiłku. Przykład usunięcia przepisu poniżej:



Opcja czwarta w menu pozwala nam wygenerować posiłki możliwe do zrobienia z podanych składników. Najpierw należy podać numery składników z wygenerowanej listy a następnie program generuje możliwe przepisy i zadaje pytanie, czy chcemy zobaczyć sposób przygotowania którejś z nich. Przykład poniżej:



Ostatnia opcja w menu, czyli numer 5 pozwala na wyjście z programu. Możliwość wyjścia z programu daje również wcześniej wspomniane menu wyświetlające się pod działaniem konkretnych segmentów programu.

5. Podsumowanie i wnioski.

Wszystkie założenia dotyczące programu zostały spełnione. Zarówno główny moduł programu, czyli generowanie posiłków, jak i dodatkowe opcje jak dodawanie przepisów, usuwanie ich oraz wyświetlanie udało się zrealizować.

Największym problemem, na który napotkano w czasie tworzenia programu, okazało się zabezpieczenie go przed niepoprawnymi wartościami podawanymi przez użytkownika. W związku z tym, że interakcja z użytkownikiem jest ważnym elementem, w licznych miejscach początkowo łatwo było doprowadzić program do niepoprawnego działania poprzez podawanie wartości w złej formie lub w ogóle niepasujących do oczekiwań.

Program ten jest przydatny dla każdego, kto nie potrafi się zdecydować, jaki posiłek zrobić – może być on dobrym rozwiązaniem dla każdego, kto chciałby stworzyć własną bazę przepisów i w prosty sposób znajdować te, które może wykonać w danej chwili.