



Akademia Łomżyńska

Wydział Nauk Informatyczno-Technologicznych

Przedmiot: Podstawy programowania – pracownia specjalistyczna

Rok akademicki: zima 2023/2024

Prowadząca: Anna Frączkowska

Kontakt: afraczkowska@al.edu.pl

Wytyczne do projektu

Projekt zaliczeniowy

Program musi być napisany w języku C zgodnym ze standardem ANSI C89 z zachowaniem zasad programowania proceduralnego oraz programowania strukturalnego.

Program powinien zawierać

- podział na pliki nagłówkowe i źródłowe
- deklaracje, definicje i użycie tablic
- deklaracje, definicje i użycie struktur
- dynamiczny przydział pamięci
- funkcje, funkcje i jeszcze raz funkcje
- obsługę plików zewnętrznych (odczyt i zapis do plików tekstowych lub binarnych)
- wskaźniki

Dodatkowo: Należy utworzyć krótką prezentację w programie PowerPoint, w której zostanie opisane i przedstawione działanie programu oraz będą w niej zawarte informacje o autorze (Imię, Nazwisko, nr albumu). Można również umieścić w prezentacji informacje o użytym kompilatorze, programie i systemie operacyjnym na którym aplikacja była testowana, ewentualnie inne przydatne informacje (np. użycie niestandardowych bibliotek, opis opcji dla aplikacji konsolowej)



Program nie powinien zawierać

- zmiennych globalnych
- bibliotek wykraczających poza standard ANSI C (w szczególności chodzi o bibliotekę `conio.h`), zapewni to przenośność programu
- długich i skomplikowanych funkcji (gdy definicja funkcji nie mieści się na ekranie to zazwyczaj znaczy, że trzeba ją podzielić na kilka mniejszych funkcji)
- instrukcji skoku `goto`

Na końcową ocenę wpływ mają

- umiejętność posługiwania składnią języka C
- poprawne użycie plików nagłówkowych i dyrektywy `#include`
- logiczny podział funkcji w plikach źródłowych, poprawne deklaracje funkcji
- umiejętność korzystania ze wskaźników
- estetyka i czytelność kodu (wcięcia, nazewnictwo funkcji i zmiennych, komentarze)
- rozwiązanie algorytmiczne
- wygoda obsługi i wygląd działającej aplikacji, (wyświetlanie menu, komunikaty dla użytkownika itp.)
- w miarę możliwości program powinien być odporny na próbę wprowadzenia niewłaściwych danych ☺

Zaliczenie

Gotowy projekt (pliki źródłowe, nagłówkowe i dokumentacja) należy podpisać według schematu: **Nazwisko_Imię_NumerIndeksu.zip** i umieścić w archiwum *.zip itp. i przesłać na platformę e-learning w wyznaczonym terminie.



Propozycje programów na zaliczenie

Tutaj można znaleźć kilka pomysłów na projekt zaliczeniowy. **Własne propozycje mile widziane.**

Statystyka

Program czyta z pliku liczby zapisane w postaci macierzy, gdzie każda kolumna zawiera wyniki pomiaru jednego z parametrów pewnego doświadczenia. Dla każdej kolumny program wyznacza wartości statystyk: suma, średnia, odchylenie standardowe, wariancja, wartość maksymalna, minimalna, mediana.

Baza danych

Prosta baza danych (np. danych osobowych, książka telefoniczna, baza filmów lub muzyki, itp.). Program powinien mieć możliwość dodawania, usuwania i wyszukiwania rekordów, zapisywania oraz odczytywania całej bazy danych z pliku. Do obsługi rekordów bazy wskazane jest wykorzystanie [listy jednokierunkowej lub dwukierunkowej](https://pl.wikipedia.org/wiki/Lista_listy_jednokierunkowej_lub_dwukierunkowej) -> <https://pl.wikipedia.org/wiki/Lista>

Analiza tekstu 1

Program po wczytaniu danego pliku tekstowego wyświetla częstości występowania każdej litery w tekście.

Wynik w postaci procentowej powinien być zapisany do pliku tekstowego (o nazwie podanej przez użytkownika) w kolejności od najczęściej do najrzadziej występującej litery. Dodatkowo program mógłby wyświetlać ilość liter, słów i inne statystyki związane z tekstem.

Analiza tekstu 2

Program stanowi rozszerzenie poprzedniego projektu. Dla danego zbioru plików z wynikami programu „Analiza tekstu 1” (najlepiej pochodzących z różnych języków) program określa podobieństwo tych statystyk do dowolnego nowego tekstu. Porównując częstości występowania liter w tekście można w ten sposób określić np. w jakim języku napisany jest dany tekst.

Słownik



Program odczytuje podany plik tekstowy i na jego podstawie tworzy słownik tzn. wynikiem działania programu powinien być plik zawierający wszystkie występujące w tekście wyrazy ułożone w kolejności alfabetycznej. Dany wyraz w słowniku może wystąpić tylko raz. Dodatkowo wyświetlana jest informacja o liczbie wystąpień każdego słowa w źródłowym tekście.

Szyfrowanie

Program szyfruje podaną treść. Może to być jeden program kodujący/dekodujący lub dwa programy, jeden szyfrujący, drugi - deszyfrujący wiadomość tekstową. Algorytm szyfrowania dowolny (kilka przykładów poniżej). Przed zaszyfrowaniem z podanego tekstu program powinien usunąć wszystkie znaki różne od liter i spacji. Przy szyfrowaniu użytkownik podaje klucz (hasło) - dowolny ciąg znaków, który jest niezbędny do późniejszego poprawnego rozszyfrowania tekstu.

Kilka przykładów algorytmów:

szyfr Cezara

Każdemu znak tekstu jawnego zastępujemy znakiem przesuniętym o określoną, stałą liczbę znaków w alfabecie. Litery z końca alfabetu stają się literami z jego początku. Przykładowo, przesuwając o 3 znaki litera A staje się literą D, litera Z staje się literą C, itd. Podane hasło przy szyfrowaniu powinno posłużyć do wygenerowania liczby całkowitej określającej o ile przesuwamy litery.

szyfr Cardano

Każda literę zastępujemy inną według ustalonego przyporządkowania, np.:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ  
QWERTYUIOPASDFGHJKLZXCVBNM
```

gdzie literę A zastępujemy literą Q, B literą W, itd. Dla alfabetu 26 znakowego mamy więc 26! możliwości różnych przyporządkowań liter. W programie należy zaproponować metodę generowania takiego przyporządkowania z podanego przy szyfrowaniu klucza (hasła).

Szyfr Vigenere'a

Jest to zmodyfikowana wersja szyfru Cezara. Kluczem jest ciąg kilku liczb. Podobnie jak w metodzie Cezara zastępujemy znaki literami przesuniętymi o pewną wartość w alfabecie, z tym, że dla każdego znaku ta wartość jest inna. Np. dla podanych w kluczu trzech liczb 2, 15 i 4 przesuwamy pierwszy znak o 2, drugi o 15, trzeci o 4, czwarty znów o 2 itd. Ponieważ każdy znak w naturalny sposób jest liczbą całkowitą w języku C (kod ASCII) nie powinno być problemu z zamianą dowolnego ciągu znaków (hasła) na ciąg liczb całkowitych wymaganych w tym algorytmie.

Macierz

Szyfrowany tekst umieszczany jest w macierzy. W zależności od podanego hasła powinna być generowana liczba określająca liczbę kolumn. Np:

T A J N

A W I



A D O M

O Ś Ć

Metoda polega na odczytaniu tekstu w kolumnach, czyli w tym przypadku otrzymamy: TAAOA DŚJWOŚNIM

XOR

Szyfrowany tekst i klucz zapisujemy w postaci binarnej. Na odpowiednich bitach tekstu i klucza wykonujemy operację XOR, czyli:

$$0 \text{ XOR } 0 = 0$$

$$0 \text{ XOR } 1 = 1$$

$$1 \text{ XOR } 0 = 1$$

$$1 \text{ XOR } 1 = 0$$

Otrzymana sekwencja bitów jest zaszyfrowanym tekstem. Odszyfrowanie sprowadza się do przeprowadzenia tej samej operacji na zaszyfrowanym tekście, gdyż jak łatwo sprawdzić

$$(A \text{ XOR } B) \text{ XOR } B = A$$

Więcej o szyfrach klasycznych (W. Guzicki) : <http://www.wiw.pl/delta/szyfry.asp>

Gra Life

Symulacja życia „bakterii” na planszy o podanych przez użytkownika rozmiarach.

Rozkład początkowy organizmów jest losowany lub może być wczytany z pliku.

Życie na planszy toczy się według następujących reguł:

- jeśli puste pole jest otoczone przez co najmniej trzy bakterie to w tym miejscu powstaje nowa bakteria
- jeżeli bakteria jest otoczona przez więcej niż trzy bakterie - ginie z braku pożywienia
- jeśli bakteria posiada ma mniej niż 2 sąsiadów - ginie z samotności

Program powinien wyświetlać kolejne tury z życia kolonii.

Zobacz: [Gra w życie](https://pl.wikipedia.org/wiki/Gra_w_%C5%BCycie) -> https://pl.wikipedia.org/wiki/Gra_w_%C5%BCycie

Master Mind

Gra polegająca na odgadnięciu sekwencji czterech z sześciu możliwych symboli (np cyfr lub liter) wylosowanych na początku gry przez komputer.

Po każdej próbie program wyświetla następujące informacje:

- liczba trafionych symboli, ale na złych pozycjach



- liczba trafionych symboli położonych w dobrym miejscu

W pliku tekstowym umieszczane są najlepsze wyniki.

Piętnastka

Program umieszcza na planszy 4×4 losowo ułożone liczby od 1 do 15 (lub znaki od „a” do „m”).

9	3		10
5	6	13	2
1	14	10	11
4	7	8	15

Celem jest w jak najmniejszej liczbie przesunięć danej liczby na wolne pole ułożyć je w porządku rosnącym.

Program łatwo można zmodyfikować tak aby działał dla planszy o dowolnej wielkości (dla chętnych).

W pliku tekstowym umieszczane są najlepsze wyniki.

Edytor tagów MP3

Dla danego pliku w formacie mp3 program pozwala wyświetlić oraz zmodyfikować zawartości etykiet (tagów) zapisanych w formacie [ID3v1](#) i [ID3v2](#) -> <https://pl.wikipedia.org/wiki/ID3>. Bardziej rozbudowany program może umożliwiać masową zamianę etykiet np. dla wszystkich utworów w podanym katalogu lub dla wszystkich utworów podanych jako argumenty programu. Odczyt tagów mp3 można również wykorzystać do stworzenia kolekcji (bazy danych) wszystkich utworów muzycznych użytkownika.

Konsolowy komunikator

Program uruchomiony na dwóch komputerach połączonych w sieci LAN pozwala na wymianę komunikatów tekstowych pomiędzy użytkownikami.