

Projekt: Binary file handling tool



**Politechnika
Śląska**

autorzy:

Marcin Przybylski

Marcin Barczyk

Mateusz Major

Hubert Szymiski

Maciej Walkowiak

Teleinformatyka sem IV, grupa TI2
prowadzący: Dr. inż Jakub Nalepa




Założenia

Przeglądanie zawartości plików binarnych jest zazwyczaj kłopotliwe i wymaga użycia narzędzi programowych, które obsługują dany format pliku. Pomocne może być opracowanie narzędzia do przeglądania takich plików o zdefiniowanym formacie z opcją eksportowania ich do pliku tekstowego.




Cel projektu






| ID | Nazwa | Opis | Faza projektu |
|----------|---|---|---------------|
| REQ01.00 | Utworzenie programu do odczytu plików binarnych i ich eksportu do plików tekstowych dowolnego formatu | Program powinien obsługiwać w obsłudze plików binarnych i ich konwersji na wybrany przez użytkownika format tekstowy | Pierwsza |
| REQ01.01 | Wybór struktury pliku wejściowego | Struktura pliku binarnego powinna być określona przez użytkownika - narzędzie powinno umożliwiać definiowanie zarówno struktur liniowych, jak i iteracyjnych (liczba iteracji zapisana w pliku) | Pierwsza |
| REQ01.02 | Lista dostępnych formatów i plików wraz z historią | Lista kilku ostatnio używanych formatów powinna być dostępna w GUI. To samo dotyczy ostatniego folderu plików. | Pierwsza |
| REQ01.03 | Prezentacja danych jako tekst oraz przechowywanie danych | Dane odczytane z pliku powinny być przedstawione jako tekst, a dane mogą być przechowywane w pliku tekstowym (całkowicie lub częściowo) | Pierwsza |
| REQ01.04 | Obsługa przetwarzania wsadowego | Program ma możliwość konwersji całego zestawu danych | Pierwsza |

| | | | |
|----------|---|--|-------|
| REQ01.05 | Czytelny  oraz przejrzyste GUI | Interfejs graficzny ma być czytelny oraz wygodny do obsługi dla każdego użytkownika niezależnie od wieku | Druga |
|----------|---|--|-------|

Wymagania funkcjonalne

| | |
|--------------------|--|
| Nazwa funkcji | select  () - wybór pliku |
| Opis | Wyświetla w nowym oknie domyślną lokalizację, np. dysk C, umożliwiając wskazanie pliku binarnego, posiada również opcję wyświetlenia lokalizacji ostatnio użytych przez użytkownika. |
| Dane wejściowe | ostatnio odwiedzane lokalizacje  . |
| Źródło danych wej. | vector<string> |
| Wynik | zwrócenie typu string z lokalizacją pliku przeznaczonego do odczytu. |
| Warunek wstępny | plik istnieje i nie jest uszkodzony. |
| Warunek końcowy | zwrócony string z lokalizacją  |
| Efekty uboczne | dodanie nowej lokalizacji do wektora ostatnio odwiedzanych folderów. |
| Powód | |

| | |
|--------------------|---|
| Nazwa funkcji | openBinaryFile() - otwarcie pliku |
| Opis | Odczytuje plik binarny do pamięci programu.  |
| Dane wejściowe | ciąg znaków bitowych |
| Źródło danych wej. | plik binarny |
| Wynik | odczytanie ciągu znaków bitowych do wektora typu string |
| Warunek wstępny | plik istnieje i jest plikiem binarnym. |
| Warunek końcowy | utworzony vect  |
| Efekty uboczne | zajęcie dodatkowego miejsca w pamięci komputera. |
| Powód | |

| | |
|--------------------|---|
| Nazwa funkcji | convertToText() - konwersja pliku  |
| Opis | Konwertuje plik binarny na plik tekstowy wybrany przez użytkownika. |
| Dane wejściowe | Ciąg znaków bitowych |
| Źródło danych wej. | Wektor z funkcji openBinaryFile() |
| Wynik | Plik tekstowy z zawartością. |
| Warunek wstępny | Istniejący plik binarny oraz wybrany format do zapisu |
| Warunek końcowy | Koniec pliku |
| Efekty uboczne | Zajęcie pamięci oraz czas trwania. |
| Powód | |

| | |
|--------------------|---|
| Nazwa funkcji | selectExt() - wybór formatu do zapisu pliku |
| Opis | Wyświetla menu zawierające dostępne formaty zapisu plików tekstowych, także ostatnio używane przez użytkownika. |
| Dane wejściowe | Lista dostępnych formatów |
| Źródło danych wej. | Vector<string> |
| Wynik | Zwrócenie typu string z wybranym formatem |
| Warunek wstępny | Wybrano format |
| Warunek końcowy | Zwrócony string |
| Efekty uboczne | Zajęcie pamięci |
| Powód | |

| | |
|--------------------|---|
| Nazwa funkcji | saveTextFile() - zapis pliku |
| Opis | Zapisuje tekst do wcześniej wybranego formatu pliku. |
| Dane wejściowe | Format pliku, lokalizacja zapisu |
| Źródło danych wej. | Zmienna typu string |
| Wynik | Zapis pliku |
| Warunek wstępny | Lokalizacja jest dostępna, plik o podanej nazwie nie istnieje przed zapisem |
| Warunek końcowy | Zapisany plik |
| Efekty uboczne | Dodanie formatu do ostatnio używanych |
| Powód | |

| | |
|--------------------|---|
| Nazwa funkcji | chooseOneOrSet() - wybór konwersji zestawu plików lub jednego. |
| Opis | Przez polecenie użytkownika wybiera konwersję zestawu plików lub jednego. |
| Dane wejściowe | Lista wyboru |
| Źródło danych wej. | Vector <string> |
| Wynik | Konwersja wybrana przez użytkownika |
| Warunek wstępny | Wybrano opcję |
| Warunek końcowy | Zwrócono string |
| Efekty uboczne | Zajętość pamięci, możliwość złego wyboru |
| Powód | |

Wymagania niefunkcjonalne

| Cecha | Metryka |
|--------------------|---|
| Szybkość | Jeden plik binarny na 10 sekund. Czas reakcji na sygnał użytkownika: 0.1 sekundy. |
| Rozmiar | Wymagana pamięć RAM: 2GB, wymagana pamięć dyskowa 1GB |
| Łatwość użycia | Czas przeszkolenia ok. 30 minut. Liczba stron systemu pomocy: 5. |
| Niezawodność | Prawdopodobieństwo błędnego wyboru skutkującego niepożądanym działaniem programu ok. 15%. Częstotliwość występowania awarii: nie częściej niż co 100 uruchomień programu. |
| Odporność na błędy | Czas restartu po awarii ok. 10 sekund |
| Przenośność | Liczba systemów docelowych: jeden. |

Diagram sekwencji

