**Dokumentacja projektu "Konwerter liczbowy".**

**Opis**

Program **„Konwerter liczbowy”** to program napisany w języku C++, który pozwala użytkownikowi na konwersję liczby dziesiętnej na trzy różne systemy liczbowe: dwójkowy, ósemkowy i szesnastkowy. Program posiada prosty interfejs, który umożliwia użytkownikowi wybór jednej z opcji, takich jak wprowadzenie nowej liczby, wybór systemu liczbowego lub zakończenie programu.

**Funkcjonalności**

Program posiada następujące funkcjonalności:

* Wprowadzenie liczby, która ma zostać zmieniona,
* Możliwość ponownego wprowadzenia liczby,
* Wybór systemu liczbowego, na który ma zostać przekonwertowana liczba,
* Wyświetlenie wyniku konwersji,
* Możliwość zakończenia programu.

**Ogólne działanie programu**

Po uruchomieniu programu, użytkownik jest powitany i musi wprowadzić liczbę z przedziału od 1 do 9999, którą chce przeliczyć. Program zawiera blokadę, która uniemożliwia wprowadzenie liczby spoza tego zakresu.

Następnie użytkownik ma możliwość wyboru, co chce zrobić z wprowadzoną liczbą. Program wyświetla menu z pięcioma opcjami:

1. Wprowadź nową liczbę.
2. Zamień podaną liczbę na system binarny (dwójkowy).
3. Zamień podaną liczbę na system oktalny (ósemkowy).
4. Zamień podaną liczbę na system heksadecymalny (szesnastkowy).
5. Zakończ program.

Użytkownik może wprowadzić liczbę odpowiadającą wybranej opcji, a następnie program wykonuje odpowiednie obliczenia i wyświetla wynik. W przypadku opcji 1, użytkownik musi wprowadzić nową liczbę z zakresu od 1 do 9999. Opcja 5 kończy działanie programu.

Program zawiera też kilka innych funkcji, które umożliwiają wyświetlanie polskich znaków oraz czyszczenie konsoli.

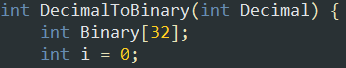
**Implementacja**

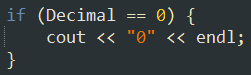
Program składa się z trzech funkcji, które odpowiadają za konwersję liczby dziesiętnej na systemy binarny, ósemkowy i szesnastkowy. Funkcje te przyjmują jako argument liczbę dziesiętną i zwracają liczbę w odpowiednim systemie liczbowym.

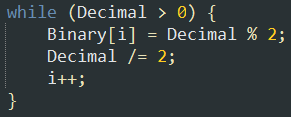
1. Funkcja **DecimalToBinary()** odpowiedzialna jest za konwersję podanej liczby dziesiętnej na jej reprezentację binarną. Algorytm polega na dzieleniu wprowadzonej liczby przez 2 i zapisywaniu kolejnych reszt w tablicy Binary. Następnie odwraca się kolejność cyfr w tablicy i wyświetla wynik.
2. Funkcja **DecimalToOctal()** odpowiedzialna jest za konwersję podanej liczby dziesiętnej na jej reprezentację oktalną. Algorytm polega na dzieleniu wprowadzonej liczby przez 8 i zapisywaniu kolejnych reszt w tablicy Octal. Następnie odwraca się kolejność cyfr w tablicy i wyświetla wynik.
3. Funkcja **DecimalToHex()** odpowiedzialna jest za konwersję podanej liczby dziesiętnej na jej reprezentację heksadecymalną. Algorytm polega na dzieleniu wprowadzonej liczby przez 2 i zapisywaniu kolejnych reszt w tablicy Hex. Dodatkowo dokonuje się konwersja liczb o wartości większej niż 9 na odpowiadające im litery alfabetu.

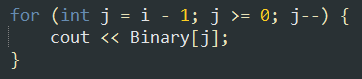
**Funkcja DecimalToBinary()**

Funkcja ta przekształca liczby dziesiętne na liczby w systemie dwójkowym. Funkcja ta przyjmuje argument typu int o nazwie Decimal.

  
Ta linijka kodu tworzy tablicę Binary o rozmiarze 32, która będzie przechowywać wartości binarne. Zmienna i jest ustawiona na 0 i będzie służyła jako licznik.

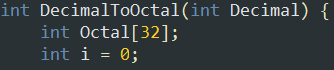
  
Jeśli liczba dziesiętna jest równa zero, funkcja wyświetla ‘0’ i kończy działanie.

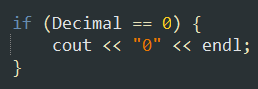
  
Pętla while będzie wykonywana do momentu, gdy wartość zmiennej Decimal jest większa od zera. Pętla przekształca liczbę dziesiętną na jej odpowiednik w systemie dwójkowym. W każdej iteracji pętli obliczana jest reszta z dzielenia przez 2, która jest przypisywana do elementu tablicy Binary o indeksie i. Następnie liczba dziesiętna jest dzielona przez 2 i zapisywana z powrotem do zmiennej Decimal. Następnie zwiększa się wartość zmiennej i.

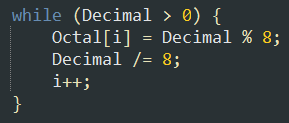
  
Pętla for wypisuje cyfry binarne z tablicy Binary w odwrotnej kolejności, zaczynając od ostatniej cyfry (na pozycji i - 1) i kończąc na pierwszej cyfrze (na pozycji 0).

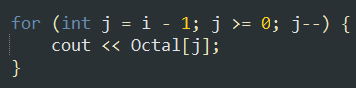
**Funkcja DecimalToOctal()**

Funkcja ta przekształca liczby dziesiętne na liczby w systemie ósemkowym. Funkcja ta przyjmuje argument typu int o nazwie Decimal.

  
Ta linijka kodu tworzy tablicę Octal o rozmiarze 32, która będzie przechowywać wartości oktalne. Zmienna i jest ustawiona na 0 i będzie służyła jako licznik.

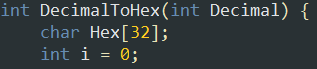
  
Jeśli liczba dziesiętna jest równa zero, funkcja wyświetla ‘0’ i kończy działanie.

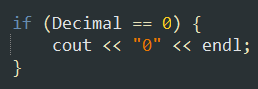
  
Pętla while będzie wykonywana do momentu, gdy wartość zmiennej Decimal jest większa od zera. Pętla przekształca liczbę dziesiętną na jej odpowiednik w systemie ósemkowym. W każdej iteracji pętli obliczana jest reszta z dzielenia przez 8, która jest przypisywana do elementu tablicy Octal o indeksie i. Następnie liczba dziesiętna jest dzielona przez 8 i zapisywana z powrotem do zmiennej Decimal. Następnie zwiększa się wartość zmiennej i.

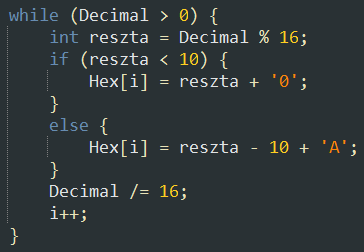
  
Pętla for wypisuje cyfry oktalne z tablicy Octal w odwrotnej kolejności, zaczynając od ostatniej cyfry (na pozycji i - 1) i kończąc na pierwszej cyfrze (na pozycji 0).

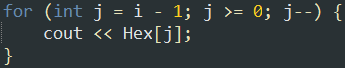
**Funkcja DecimalToHex()**

Funkcja ta przekształca liczby dziesiętne na liczby w systemie szesnastkowym. Funkcja ta przyjmuje argument typu int o nazwie Decimal.

  
Ta linijka kodu tworzy tablicę Hex o rozmiarze 32, która będzie przechowywać wartości hexadecymalne. Zmienna i jest ustawiona na 0 i będzie służyła jako licznik.

  
Jeśli liczba dziesiętna jest równa zero, funkcja wyświetla ‘0’ i kończy działanie.

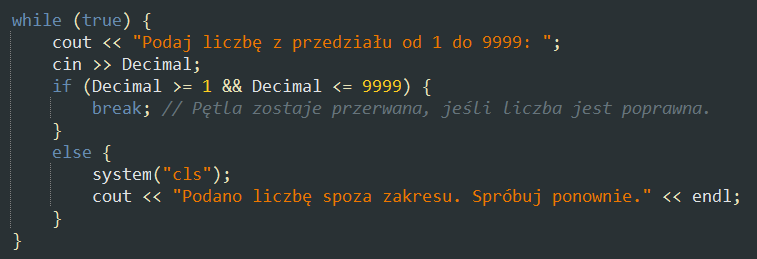
  
Pętla while będzie wykonywana do momentu, gdy wartość zmiennej Decimal jest większa od zera. Pętla przekształca liczbę dziesiętną na jej odpowiednik w systemie szesnastkowym. W każdej iteracji pętli obliczana jest reszta z dzielenia przez 16. Następnie, jeśli reszta jest mniejsza niż 10, dodaje do tablicy Hex wartość ‘0’ plus resztę. W przeciwnym przypadku do tablicy Hex dodaje wartość ‘A’ pomniejszoną o 10 plus resztę, co daje odpowiednią literę dla liczby szesnastkowej. Następnie liczba dziesiętna jest dzielona przez 16. Następnie zwiększa się wartość zmiennej i.

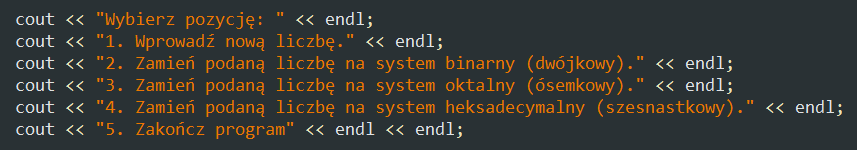
  
Pętla for wypisuje cyfry hexadecymalne z tablicy Hex w odwrotnej kolejności, zaczynając od ostatniej cyfry (na pozycji i - 1) i kończąc na pierwszej cyfrze (na pozycji 0).

**Funkcja główna main()**

  
Pierwsza linijka kodu zawiera funkcję setlocale, która ustawia kodowanie znaków na "Polish", aby umożliwić wyświetlanie polskich znaków na konsoli.

  
Zmienna Decimal, która przechowuje liczbę wprowadzoną przez użytkownika.

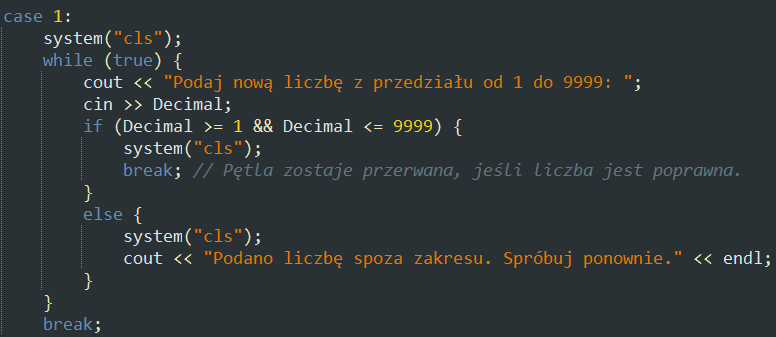
  
Pętla while będzie kontynuować wykonywanie kodu, dopóki warunek logiczny "true" jest prawdziwy. Pętla pobierze od użytkownika liczbę z zakresu 1 – 9999. Jeśli wartość podana przez użytkownika się zgadza, pętla zostanie przerwana za pomocą instrukcji break, a program przejdzie do następnej części kodu poza pętlą while. Jeśli wartość podana przez użytkownika jest poza tym zakresem, pętla będzie kontynuować wykonywanie, do momentu aż użytkownik wprowadzi prawidłową wartość.

Następnie program wyświetla menu, w którym użytkownik może wybrać jedną z pięciu opcji: wprowadzenie nowej liczby, konwersję na system binarny, konwersję na system oktalny, konwersję na system heksadecymalny lub zamknięcie programu.

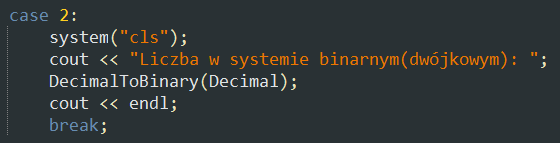
  
Zmienna przechowująca liczbę, która odpowiada za wybieranie pozycji w programie.

  
Następnie, program używa instrukcji switch do wyboru odpowiedniej opcji w zależności od wyboru użytkownika. Każda opcja wywołuje funkcję odpowiedzialną za konwersję liczby na wybrany system i wyświetlenie wyniku.

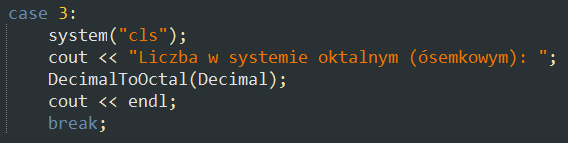
**case 1:**

  
Pierwszy case odpowiedzialny jest za ponowne wprowadzenie liczby do zmiennej Decimal. Tak jak wcześniej pętla while blokuje wprowadzanie liczb spoza zakresu  
1 – 9999.

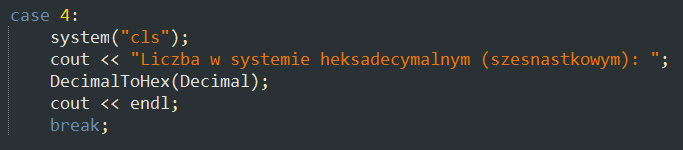
**case 2:**

  
Drugi case odpowiedzialny jest za wywołanie funkcji DecimalToBinary(Decimal). Funkcja pobiera liczbę dziesiętną i konwertuje ją na system binarny już na początku działania programu. Następnie wyświetla tą liczbę po wywołaniu funkcji.

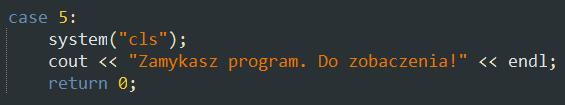
**case 3:**

  
Trzeci case odpowiedzialny jest za wywołanie funkcji DecimalToOctal(Decimal). Funkcja pobiera liczbę dziesiętną i konwertuje ją na system oktalny już na początku działania programu. Następnie wyświetla tą liczbę po wywołaniu funkcji.

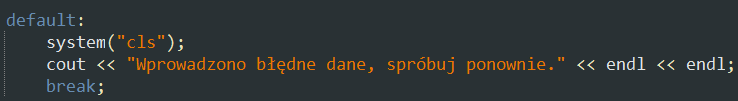
**case 4:**

  
Czwarty case odpowiedzialny jest za wywołanie funkcji DecimalToHex(Decimal). Funkcja pobiera liczbę dziesiętną i konwertuje ją na system heksadecymalny już na początku działania programu. Następnie wyświetla tą liczbę po wywołaniu funkcji.

**case 5:**

  
Ostatni case odpowiedzialny jest za zamknięcie programu. Po wybraniu tej opcji dostajemy wiadomość pożegnalną a następnie program kończy swoje działanie.

**default:**

  
Wartość defaultowa – jeśli użytkownik wprowadzi liczbę spoza zakresu 1 – 5 (te liczby to numery poszczególnych opcji menu) wyświetli się komunikat o błędnym wprowadzeniu danych i program poprosi o ponowne wybranie opcji.