**Dokumentacja projektu „Waluta”**

**Autorzy:** Lisowski Filip & Wojsa Jakub

**Spis treści**

[**Opis** 3](#_Toc134141571)

[**Funkcjonalności** 3](#_Toc134141572)

[**Ogólne działanie programu** 3](#_Toc134141573)

[**Implementacja** 4](#_Toc134141574)

[**Funkcja CurrencyToCurrency()** 5](#_Toc134141575)

[**Warunek pierwszy** 6](#_Toc134141576)

[**Warunek drugi** 8](#_Toc134141577)

[**Warunek trzeci** 10](#_Toc134141578)

[**Funkcja Exchange()** 12](#_Toc134141579)

[**Warunek pierwszy** 12](#_Toc134141580)

[**Warunek drugi** 15](#_Toc134141581)

[**Warunek trzeci** 17](#_Toc134141582)

[**Funkcja główna main()** 19](#_Toc134141583)

[**Pobieranie kwoty użytkownika** 20](#_Toc134141584)

[**Pobranie waluty użytkownika** 21](#_Toc134141585)

[**MENU użytkownika** 23](#_Toc134141586)

[**Aktualne saldo użytkownika** 23](#_Toc134141587)

[**Wyświetlenie zawartości menu** 24](#_Toc134141588)

[**Zawartość menu - instrukcja warunkowa switch** 25](#_Toc134141589)

[**Warunek pierwszy** 25](#_Toc134141590)

[**Warunek drugi** 27](#_Toc134141591)

[**Warunek trzeci** 28](#_Toc134141592)

[**Warunek czwarty** 28](#_Toc134141593)

# **Opis**

Program **„Waluty”** to program napisany w języku C++, który pozwala użytkownikowi na przewalutowanie wprowadzonej sumy pieniędzy w środowisku PLN, EUR, USD. Program posiada prosty interfejs, który umożliwia użytkownikowi wybór jednej z opcji, takich jak wprowadzenie nowej sumy pieniędzy, zmiana waluty, przewalutowanie, rozmienienie pieniędzy na banknoty i monety o nominałach 10, 5, 2, 1 lub zakończenie programu.

# **Funkcjonalności**

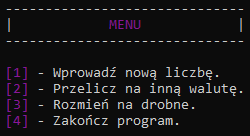
Program posiada następujące funkcjonalności:

* Wprowadzenie sumy pieniędzy,
* Wybranie interesującej waluty,
* Możliwość ponownego wprowadzenia sumy pieniędzy i wybrania interesującej waluty,
* Wyświetlenie wyniku zmiany salda,
* Możliwość zakończenia programu

# **Ogólne działanie programu**

Po uruchomieniu programu, użytkownik jest powitany i poproszony o wprowadzenie liczby z przedziału od 1 do 999999, którą chce zmienić. Program zawiera blokadę, która uniemożliwia wprowadzenie liczby spoza tego zakresu. Następnie użytkownik musi wybrać interesującą go walutę żeby przejść dalej.

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



Użytkownik może wprowadzić liczbę odpowiadającą wybranej opcji, a następnie program wyświetla wyniki. W przypadku opcji 1, użytkownik musi wprowadzić nową liczbę z zakresu od 1 do 999999. Opcja 4 kończy działanie programu.

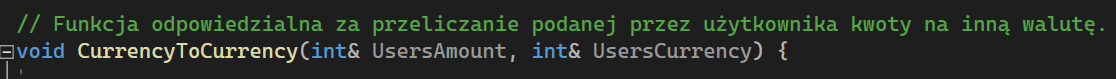
Program zawiera też kilka innych funkcji, które umożliwiają wyświetlanie polskich znaków i kolorowych napisów oraz funkcja czyszczenia okna konsoli.

# **Implementacja**

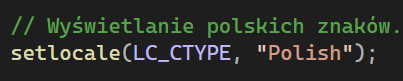
Program składa się z dwóch funkcji, które odpowiadają za przewalutowanie kwoty lub rozmienienie pieniędzy użytkownika. Funkcje te przyjmują jako argument liczbę dziesiętną i zwracają wyniki poprzez odpowiednie obliczenia.

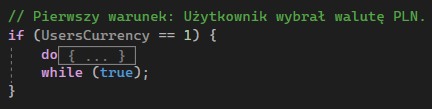
1. Funkcja **CurrencyToCurrency()** odpowiedzialna jest za przeliczanie kwoty podanej przez użytkownika w jednej walucie na jej odpowiednik w innej walucie. Funkcja przyjmuje dwa parametry wejściowe: UsersAmount (kwota podana przez użytkownika) i UsersCurrency (waluta, w której użytkownik podał kwotę). Funkcja zawiera dwie pętle do...while (dla każdej waluty odpowiednio) oraz instrukcje warunkowe if...else if w celu wyboru właściwej konwersji. W przypadku wybrania złej opcji w menu, funkcja wyświetla komunikat o błędzie i prosi o ponowny wybór.
2. Funkcja **Exchange()** odpowiedzialna jest za rozmienianie kwoty podanej przez użytkownika na banknoty i monety o nominałach 10, 5, 2, 1. Funkcja przyjmuje dwa parametry wejściowe: UsersAmount (kwota podana przez użytkownika) i UsersCurrency (waluta, w której użytkownik podał kwotę). Funkcja zawiera instrukcje warunkowe if...else if w celu wyboru właściwej konwersji.

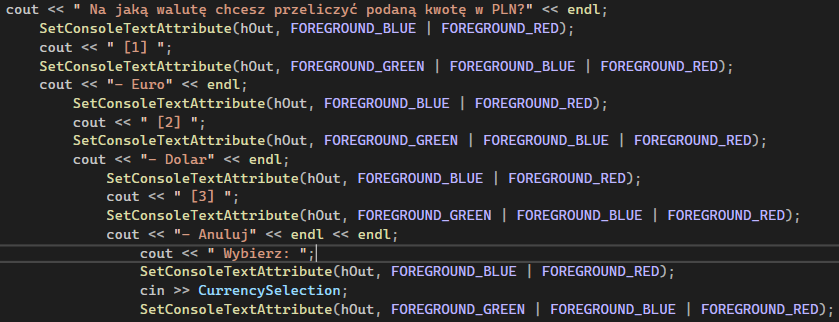
# **Funkcja CurrencyToCurrency()**

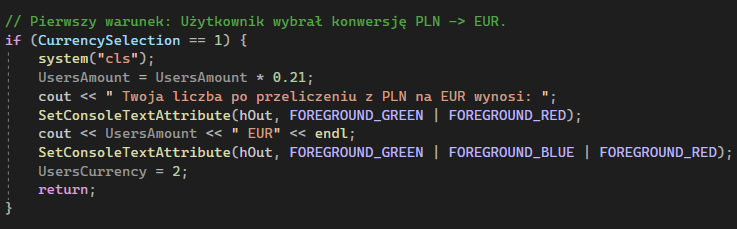
 Funkcja ta zamienia walutę podaną przez użytkownika na jej odpowiednik w innej walucie. Funkcja przyjmuje argumenty typu int o nazwie UsersCurrency i UsersCurrency.

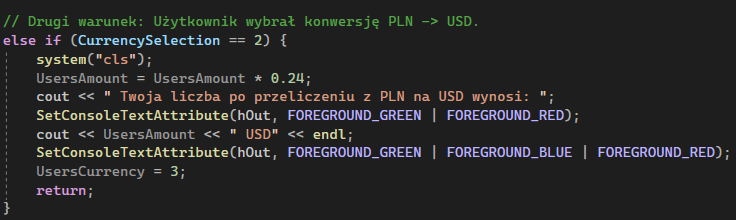
  
Zmienna przechowująca liczbę odpowiadającą za wybieranie pozycji w menu.

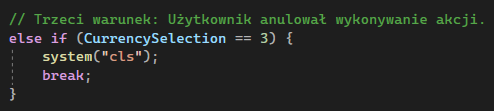
  
Wyświetlanie polskich znaków.

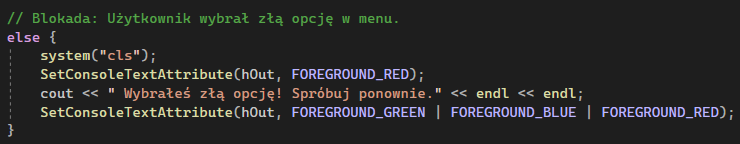
**Warunek pierwszy**  
  
  
Jeśli użytkownik wybrał walutę PLN program wykonuje zmianę waluty z PLN na EUR lub USD.

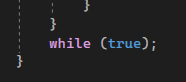
  
Te linijki wyświetlają na ekranie pytanie dla użytkownika o wybór waluty. Linie używają funkcji SetConsoleTextAttribute do zmiany kolorów tekstu w konsoli. Program wyświetla trzy opcje: [1] - Euro, [2] - Dolar i [3] - Anuluj. Następnie program prosi użytkownika o wybór opcji przez wyświetlenie tekstu "Wybierz: " i odczytuje wartość wybranej opcji z konsoli za pomocą cin i zapisuje ją do zmiennej CurrencySelection. Następnie program sprawdza wybraną opcję za pomocą instrukcji warunkowej.

  
Jeśli użytkownik wybierze opcję [1], to wartość podanej kwoty w PLN jest przeliczana na EUR, mnożąc ją przez 0,21 (kurs walutowy PLN – EUR), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z PLN na EUR wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘2’ aby kwota użytkownika była wyświetlana w EUR. Następnie kończy działanie poprzez instrukcję return.

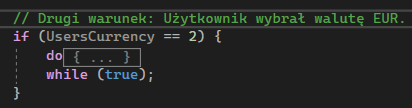
  
Jeśli użytkownik wybierze opcję [2], to wartość podanej kwoty w PLN jest przeliczana na USD, mnożąc ją przez 0,24 (kurs walutowy PLN – USD), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z PLN na USD wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘3’ aby kwota użytkownika była wyświetlana w USD. Następnie kończy działanie funkcji poprzez instrukcję return.

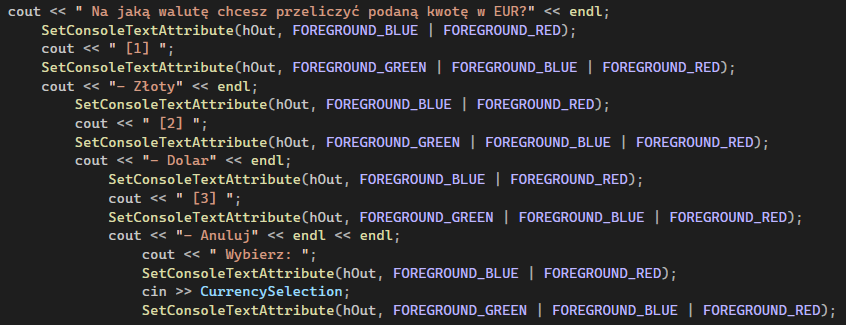
  
Jeśli użytkownik wybierze opcję [3], program czyści konsolę i działanie funkcji jest przerywane poprzez instrukcję break.

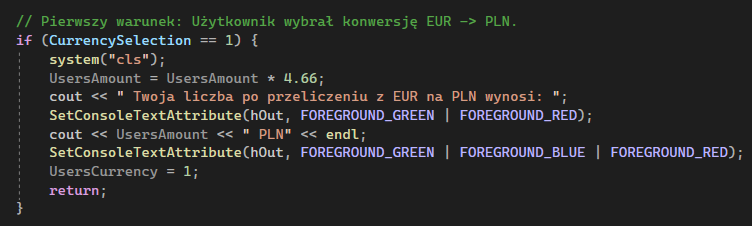
  
Jeśli użytkownik wybierze inną opcję niż [1], [2] lub [3], to na ekranie pojawi się komunikat o błędzie i kod poprosi użytkownika o ponowne wybranie opcji.

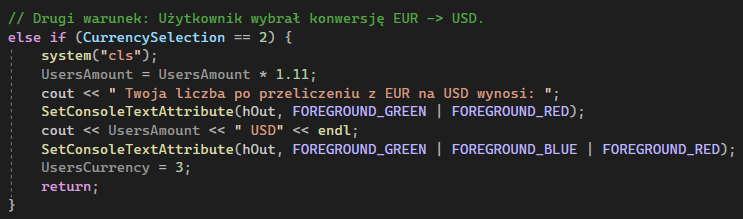
  
Ostatnia linia to zamknięcie pętli do…while przy użyciu instrukcji while(true). Oznacza to, że pętla będzie wykonywała się w nieskończoność, aż użytkownik nie wybierze jednej z trzech opcji: [1], [2] lub [3].

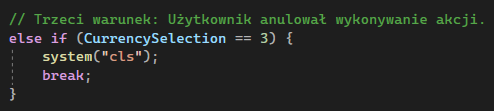
## **Warunek drugi**

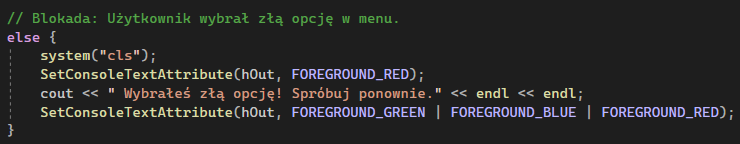
  
Jeśli użytkownik wybrał walutę EUR program wykonuje zmianę waluty z EUR na PLN lub USD.

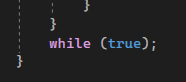
Te linijki wyświetlają na ekranie pytanie dla użytkownika o wybór waluty. Linie używają funkcji SetConsoleTextAttribute do zmiany kolorów tekstu w konsoli. Program wyświetla trzy opcje: [1] - Złoty, [2] - Dolar i [3] - Anuluj. Następnie program prosi użytkownika o wybór opcji przez wyświetlenie tekstu "Wybierz: " i odczytuje wartość wybranej opcji z konsoli za pomocą cin i zapisuje ją do zmiennej CurrencySelection. Następnie program sprawdza wybraną opcję za pomocą instrukcji warunkowej.

  
Jeśli użytkownik wybierze opcję [1], to wartość podanej kwoty w EUR jest przeliczana na PLN, mnożąc ją przez 4,66 (kurs walutowy EUR – PLN), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z EUR na PLN wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘1’ aby kwota użytkownika była wyświetlana w PLN. Następnie kończy działanie poprzez instrukcję return.

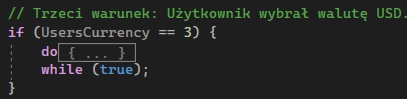
  
Jeśli użytkownik wybierze opcję [2], to wartość podanej kwoty w EUR jest przeliczana na USD, mnożąc ją przez 1,11 (kurs walutowy EUR – USD), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z EUR na USD wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘3’ aby kwota użytkownika była wyświetlana w USD. Następnie kończy działanie funkcji poprzez instrukcję return.

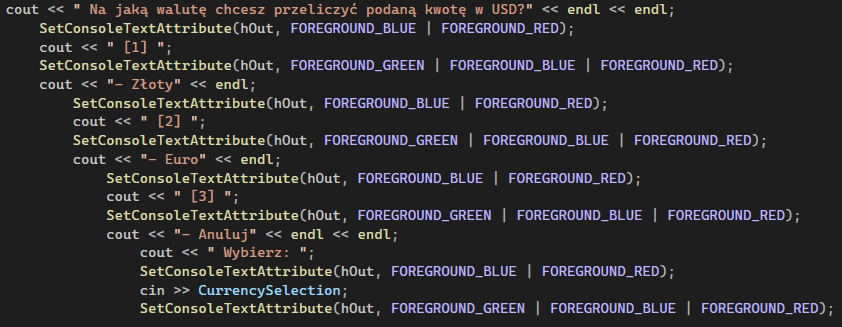
  
Jeśli użytkownik wybierze opcję [3], program czyści konsolę i działanie funkcji jest przerywane poprzez instrukcję break.

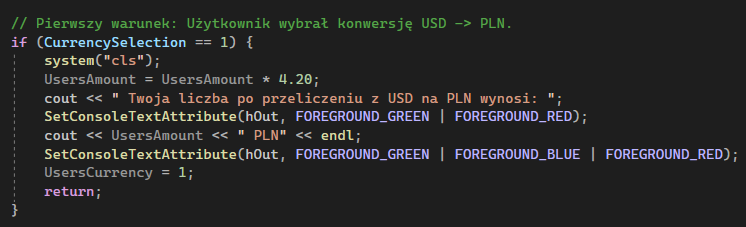
  
Jeśli użytkownik wybierze inną opcję niż [1], [2] lub [3], to na ekranie pojawi się komunikat o błędzie i kod poprosi użytkownika o ponowne wybranie opcji.

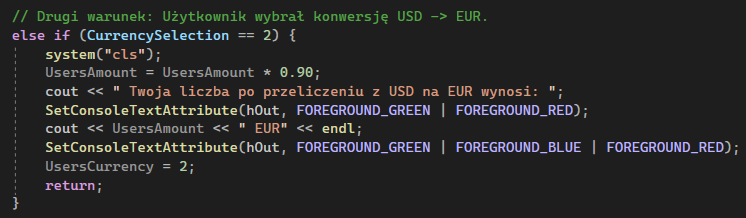
  
Ostatnia linia to zamknięcie pętli do…while przy użyciu instrukcji while(true). Oznacza to, że pętla będzie wykonywała się w nieskończoność, aż użytkownik nie wybierze jednej z trzech opcji: [1], [2] lub [3].

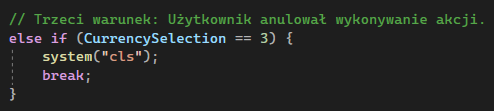
## **Warunek trzeci**

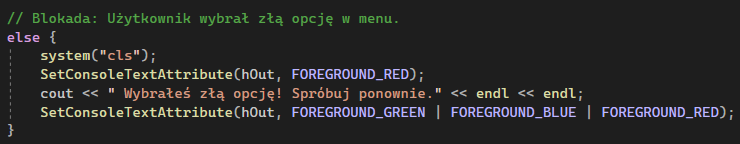
  
Jeśli użytkownik wybrał walutę USD program wykonuje zmianę waluty z USD na PLN lub EUR.

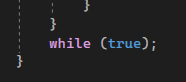
  
Te linijki wyświetlają na ekranie pytanie dla użytkownika o wybór waluty. Linie używają funkcji SetConsoleTextAttribute do zmiany kolorów tekstu w konsoli. Program wyświetla trzy opcje: [1] - Złoty, [2] - Euro i [3] - Anuluj. Następnie program prosi użytkownika o wybór opcji przez wyświetlenie tekstu "Wybierz: " i odczytuje wartość wybranej opcji z konsoli za pomocą cin i zapisuje ją do zmiennej CurrencySelection. Następnie program sprawdza wybraną opcję za pomocą instrukcji warunkowej.

  
Jeśli użytkownik wybierze opcję [1], to wartość podanej kwoty w USD jest przeliczana na PLN, mnożąc ją przez 4,20 (kurs walutowy USD – PLN), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z USD na PLN wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘1’ aby kwota użytkownika była wyświetlana w PLN. Następnie kończy działanie poprzez instrukcję return.

  
Jeśli użytkownik wybierze opcję [2], to wartość podanej kwoty w USD jest przeliczana na PLN, mnożąc ją przez 0,90 (kurs walutowy USD – EUR), a wynik jest wyświetlany na ekranie z opisem "Twoja liczba po przeliczeniu z USD na EUR wynosi: ". Następuje przypisanie do zmiennej UsersCurrency wartości ‘2’ aby kwota użytkownika była wyświetlana w EUR. Następnie kończy działanie poprzez instrukcję return.

  
Jeśli użytkownik wybierze opcję [3], program czyści konsolę i działanie funkcji jest przerywane poprzez instrukcję break.

  
Jeśli użytkownik wybierze inną opcję niż [1], [2] lub [3], to na ekranie pojawi się komunikat o błędzie i kod poprosi użytkownika o ponowne wybranie opcji.

  
Ostatnia linia to zamknięcie pętli do…while przy użyciu instrukcji while(true). Oznacza to, że pętla będzie wykonywała się w nieskończoność, aż użytkownik nie wybierze jednej z trzech opcji: [1], [2] lub [3].

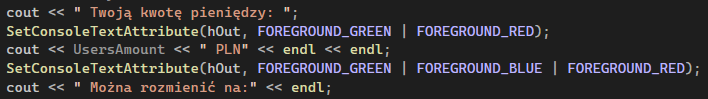
# **Funkcja Exchange()**

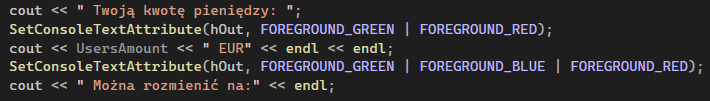
  
Funkcja ta odpowiedzialna jest za rozmienianie podanej przez użytkownika kwoty na banknoty i monety o nominałach 10, 5, 2, 1. Następnie po rozmienieniu wyświetla na ekranie konsoli wynik.

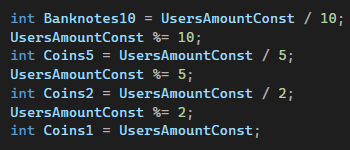
  
Ta zmienna poprzez przypisanie do niej innej zmiennej odpowiedzialna jest za przeliczanie podanej przez użytkownika kwoty nie ingerując w zmienną odpowiedzialną za przechowywanie kwoty użytkownika.

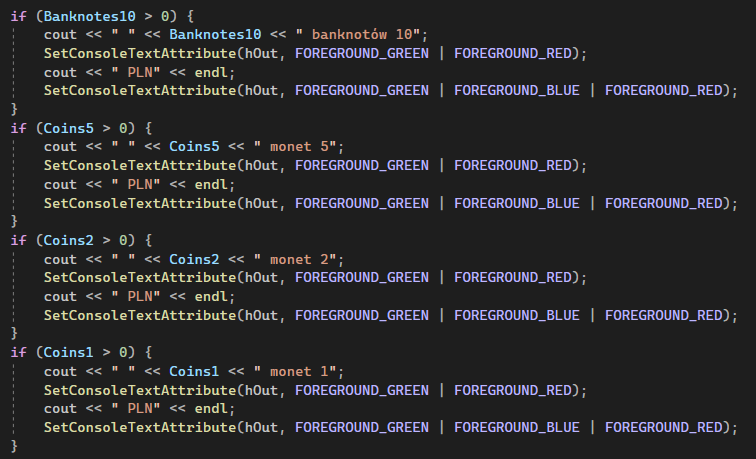
## **Warunek pierwszy**

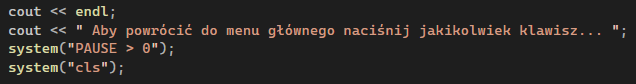
  
Pierwszy warunek odpowiedzialny jest za rozmienianie kwoty użytkownika podanej w PLN. Jeśli użytkownik wybrał PLN zadziała ten warunek.

****Wyświetlany jest tekst " Twoją kwotę pieniędzy: ", zawartość zmiennej UsersAmount i   
" PLN". Następnie wyświetlany jest tekst " Można rozmienić na: ".

  
Ten fragment służy do wyświetlenia użytkownikowi informacji na temat kwoty pieniędzy, którą wprowadził.

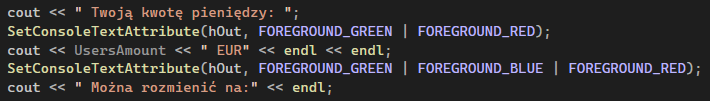
  
Teraz następuje dzielenie podanej przez użytkownika kwoty. Kod dzieli wartość tej zmiennej przez różne liczby (10, 5, 2 i 1) i przypisuje wyniki do zmiennych Banknotes10, Coins5, Coins2 i Coins1 odpowiednio. Pierwsza linia kodu dzieli wartość zmiennej UsersAmountConst przez 10 i przypisuje wynik do zmiennej Banknotes10. Zmienna ta reprezentuje ilość banknotów o nominale 10. Druga linia kodu aktualizuje wartość zmiennej UsersAmountConst poprzez przypisanie reszty z dzielenia przez 10. W ten sposób zmienna UsersAmountConst zawiera pozostałą kwotę, która nie została uwzględniona w poprzednim kroku. Kolejne linie kodu wykonują podobne operacje dla monet o nominałach 5, 2 i 1. Ostatecznie zmienna Coins1 reprezentuje ilość monet o nominale 1.

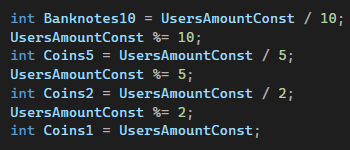
  
Ten fragment kodu wyświetla wyniki obliczeń z poprzedniego fragmentu kodu, określając ile jest banknotów i monet danego nominału. Jeśli zmienna Banknotes10 jest większa niż zero, to program wyświetli tekst "***x*** banknotów 10 PLN", gdzie ***x*** to wartość zmiennej Banknotes10. Następnie jeśli zmienna Coins5 jest większa niż zero, to program wyświetli tekst "***x*** monet 5 PLN", a jeśli zmienna Coins2 lub Coins1 jest większa niż zero, program wyświetli odpowiednio "***x*** monet 2 PLN" lub "***x*** monet 1 PLN".

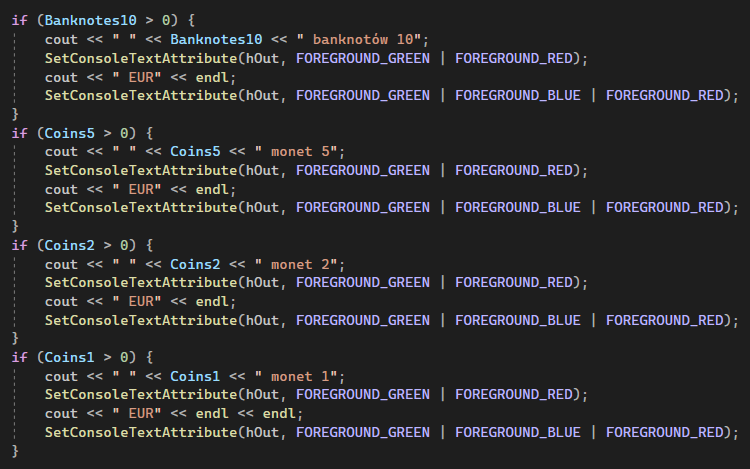
  
Program wyświetla tekst "Aby powrócić do menu głównego naciśnij jakikolwiek klawisz... ". Instrukcja system("PAUSE > 0") zatrzymuje działanie programu i czeka na dowolny klawisz, który zostanie naciśnięty przez użytkownika. Następnie, system("cls") czyści zawartość konsoli, czyli usuwa wyświetlone wcześniej wiadomości i wyniki obliczeń.

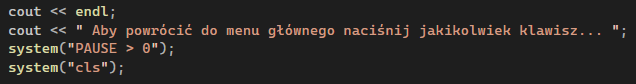
## **Warunek drugi**

  
Drugi warunek działa na takiej samej zasadzie co poprzedni. Jest odpowiedzialny za rozmienianie kwoty użytkownika podanej w EUR. Jeśli użytkownik wybrał EUR zadziała ten warunek.

Ten fragment służy do wyświetlenia użytkownikowi informacji na temat kwoty pieniędzy, którą wprowadził.

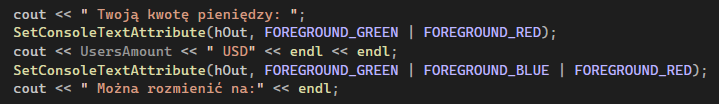
  
Teraz następuje dzielenie podanej przez użytkownika kwoty. Kod dzieli wartość tej zmiennej przez różne liczby (10, 5, 2 i 1) i przypisuje wyniki do zmiennych Banknotes10, Coins5, Coins2 i Coins1 odpowiednio. Pierwsza linia kodu dzieli wartość zmiennej UsersAmountConst przez 10 i przypisuje wynik do zmiennej Banknotes10. Zmienna ta reprezentuje ilość banknotów o nominale 10. Druga linia kodu aktualizuje wartość zmiennej UsersAmountConst poprzez przypisanie reszty z dzielenia przez 10. W ten sposób zmienna UsersAmountConst zawiera pozostałą kwotę, która nie została uwzględniona w poprzednim kroku. Kolejne linie kodu wykonują podobne operacje dla monet o nominałach 5, 2 i 1. Ostatecznie zmienna Coins1 reprezentuje ilość monet o nominale 1.

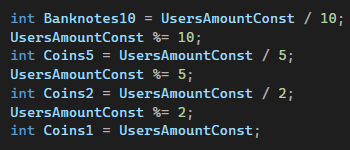
  
Ten fragment kodu wyświetla wyniki obliczeń z poprzedniego fragmentu kodu, określając ile jest banknotów i monet danego nominału. Jeśli zmienna Banknotes10 jest większa niż zero, to program wyświetli tekst "***x*** banknotów 10 EUR", gdzie ***x*** to wartość zmiennej Banknotes10. Następnie jeśli zmienna Coins5 jest większa niż zero, to program wyświetli tekst "***x*** monet 5 EUR", a jeśli zmienna Coins2 lub Coins1 jest większa niż zero, program wyświetli odpowiednio "***x*** monet 2 EUR" lub "***x*** monet 1 EUR".

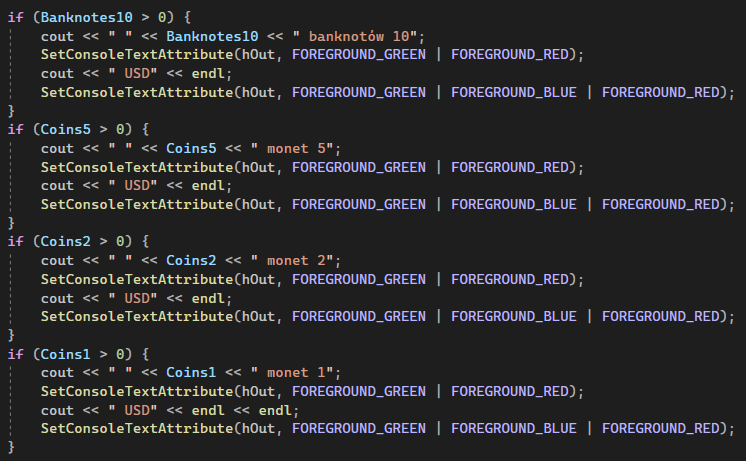
  
Program wyświetla tekst "Aby powrócić do menu głównego naciśnij jakikolwiek klawisz... ". Instrukcja system("PAUSE > 0") zatrzymuje działanie programu i czeka na dowolny klawisz, który zostanie naciśnięty przez użytkownika. Następnie, system("cls") czyści zawartość konsoli, czyli usuwa wyświetlone wcześniej wiadomości i wyniki obliczeń.

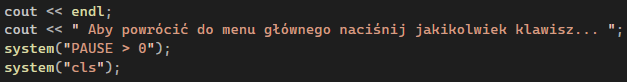
## **Warunek trzeci**

  
Drugi warunek działa na takiej samej zasadzie co warunek pierwszy i drugi. Jest odpowiedzialny za rozmienianie kwoty użytkownika podanej w USD. Jeśli użytkownik wybrał USD zadziała ten warunek.

  
Ten fragment służy do wyświetlenia użytkownikowi informacji na temat kwoty pieniędzy, którą wprowadził.

  
Teraz następuje dzielenie podanej przez użytkownika kwoty. Kod dzieli wartość tej zmiennej przez różne liczby (10, 5, 2 i 1) i przypisuje wyniki do zmiennych Banknotes10, Coins5, Coins2 i Coins1 odpowiednio. Pierwsza linia kodu dzieli wartość zmiennej UsersAmountConst przez 10 i przypisuje wynik do zmiennej Banknotes10. Zmienna ta reprezentuje ilość banknotów o nominale 10. Druga linia kodu aktualizuje wartość zmiennej UsersAmountConst poprzez przypisanie reszty z dzielenia przez 10. W ten sposób zmienna UsersAmountConst zawiera pozostałą kwotę, która nie została uwzględniona w poprzednim kroku. Kolejne linie kodu wykonują podobne operacje dla monet o nominałach 5, 2 i 1. Ostatecznie zmienna Coins1 reprezentuje ilość monet o nominale 1.

Ten fragment kodu wyświetla wyniki obliczeń z poprzedniego fragmentu kodu, określając ile jest banknotów i monet danego nominału. Jeśli zmienna Banknotes10 jest większa niż zero, to program wyświetli tekst "***x*** banknotów 10 USD", gdzie ***x*** to wartość zmiennej Banknotes10. Następnie jeśli zmienna Coins5 jest większa niż zero, to program wyświetli tekst "***x*** monet 5 USD ", a jeśli zmienna Coins2 lub Coins1 jest większa niż zero, program wyświetli odpowiednio "***x*** monet 2 USD " lub "***x*** monet 1 USD ".

  
Program wyświetla tekst "Aby powrócić do menu głównego naciśnij jakikolwiek klawisz... ". Instrukcja system("PAUSE > 0") zatrzymuje działanie programu i czeka na dowolny klawisz, który zostanie naciśnięty przez użytkownika. Następnie, system("cls") czyści zawartość konsoli, czyli usuwa wyświetlone wcześniej wiadomości i wyniki obliczeń.

# **Funkcja główna main()**

  
Wyświetlanie polskich znaków.

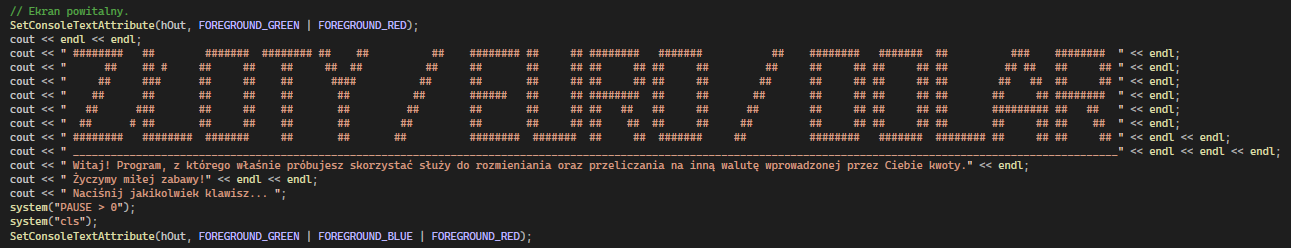
  
Zmienna przechowująca podaną przez użytkownika kwotę.

  
Zmienna przechowująca podaną przez użytkownika walutę.

  
Zmienna odpowiedzialna za otwieranie podanej przez użytkownika pozycji menu.

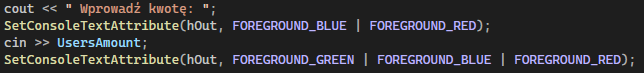
  
Kod ten maksymalizuje okno konsoli do pełnej wielkości ekranu. Pozwala to na lepszą widoczność i wygodne korzystanie z aplikacji.

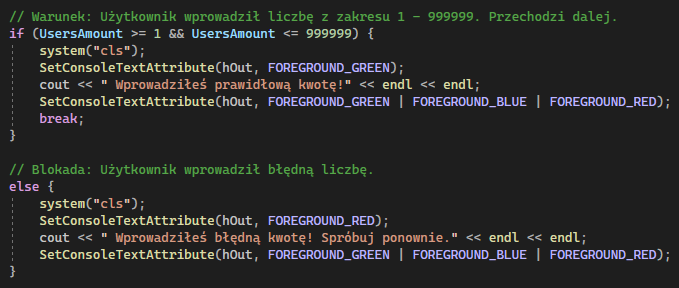
  
Pobieranie uchwytu (handle) do standardowego urządzenia wyjściowego, które w tym przypadku oznacza ekran konsoli.

  
Ekran powitalny programu. Niżej program wyświetla tekst " Naciśnij jakikolwiek klawisz... ". Instrukcja system("PAUSE > 0") zatrzymuje działanie programu i czeka na dowolny klawisz, który zostanie naciśnięty przez użytkownika. Następnie, system("cls") czyści zawartość konsoli, czyli usuwa wyświetlone wcześniej wiadomości i wyniki obliczeń.

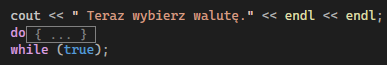
## **Pobieranie kwoty użytkownika**

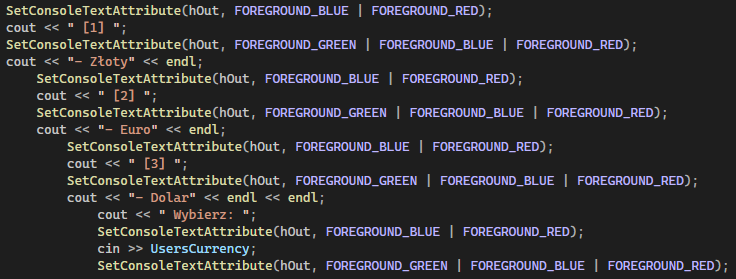
  
Instrukcja do...while odpowiedzialna za wprowadzenie odpowiedniej kwoty.

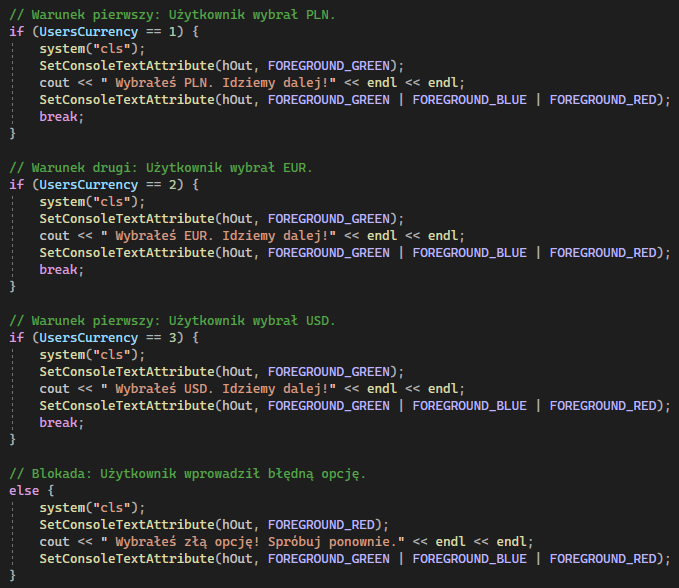
  
Pobranie wartości od użytkownika i przypisanie jej do zmiennej UsersAmount.

  
Sprawdzenie, czy użytkownik wprowadził poprawną liczbę (1 - 999999). Jeśli liczba jest poprawna, zostanie wyświetlony komunikat "Wprowadziłeś prawidłową kwotę!", a jeśli liczba jest niepoprawna, zostanie wyświetlony komunikat "Wprowadziłeś błędną kwotę! Spróbuj ponownie".

## **Pobranie waluty użytkownika**

  
Instrukcja do...while odpowiedzialna za wybranie przez użytkownika interesującej go waluty.

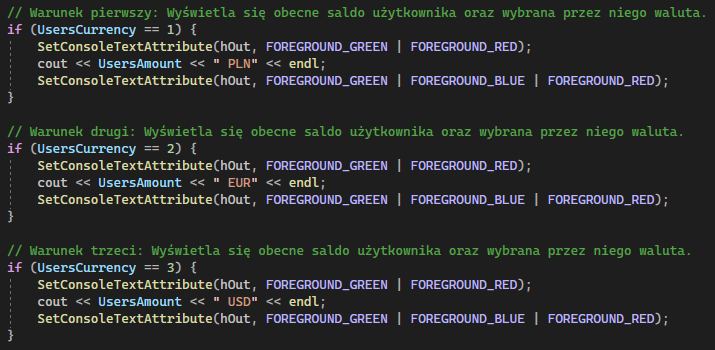
  
Wyświetlanie listy wyboru. Użytkownik może wybrać jedną z trzech opcji: Złoty, Euro, Dolar. Wartość przypisywana jest do zmiennej UsersCurrency.

  
Sprawdzenie, jaką walutę wybrał użytkownik. Jeśli użytkownik wybrał PLN (wartość 1), zostanie wyświetlony komunikat "Wybrałeś PLN. Idziemy dalej!". Jeśli użytkownik wybrał EUR (wartość 2), zostanie wyświetlony komunikat "Wybrałeś EUR. Idziemy dalej!". Jeśli użytkownik wybrał USD (wartość 3), zostanie wyświetlony komunikat "Wybrałeś USD. Idziemy dalej!". Jeśli użytkownik wprowadził nieprawidłową wartość, na przykład inną niż 1, 2 lub 3, zostanie wyświetlony komunikat "Wybrałeś złą opcję! Spróbuj ponownie." oraz zostanie poproszony o ponowne wprowadzenie odpowiedniej opcji.

## **MENU użytkownika**

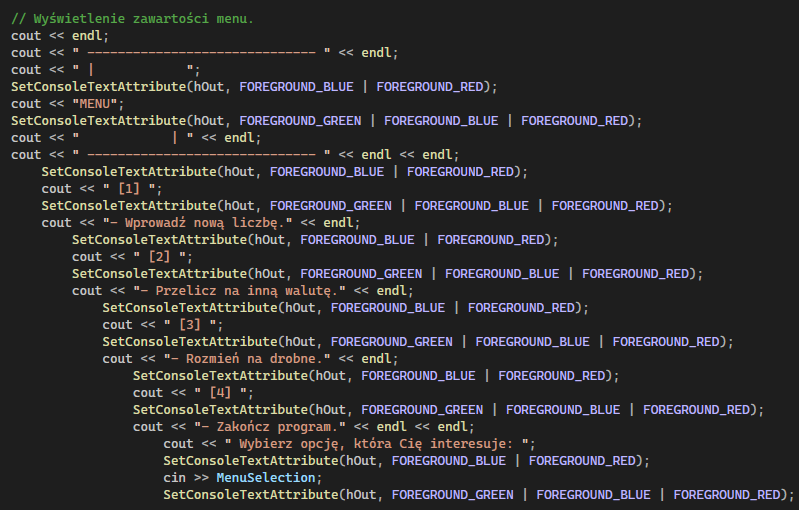
  
Instrukcja do...while odpowiedzialna za wyświetlenie obecnego salda użytkownika oraz zawartości menu.

### **Aktualne saldo użytkownika**

  
Wyświetlenie aktualnego salda użytkownika oraz wybranej przez niego waluty. Jeśli użytkownik wybrał PLN (wartość 1), zostanie wyświetlone jego saldo w PLN. Jeśli użytkownik wybrał EUR (wartość 2), zostanie wyświetlone jego saldo w EUR. Jeśli użytkownik wybrał USD (wartość 3), zostanie wyświetlone jego saldo w USD.

### 

### **Wyświetlenie zawartości menu**

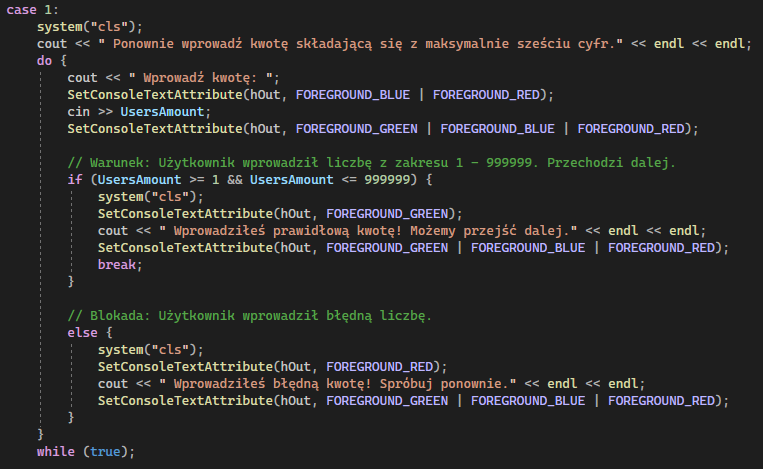
  
Wyświetlanie użytkownikowi zawartości menu, w którym może wybrać jedną z czterech opcji, aby zarządzać swoim saldem i walutą. Na początku widnieje napis "MENU". Następnie wyświetla się lista czterech opcji do wyboru, numerowanych od 1 do 4.   
Opcja pierwsza pozwala użytkownikowi wprowadzić nową wartość swojego salda. Opcja druga pozwala przeliczyć saldo na inną walutę. Opcja trzecia pozwala użytkownikowi rozmienić swoje saldo na mniejsze nominały, a opcja czwarta zamyka program. Na końcu kod prosi użytkownika o wybór opcji, którą chce wybrać, i pobiera wprowadzoną przez użytkownika wartość do zmiennej MenuSelection.

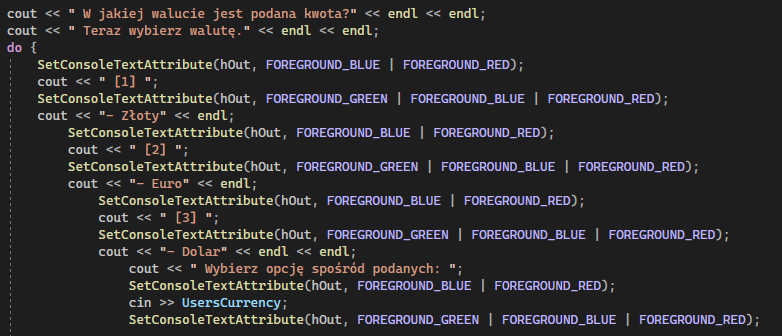
## **Zawartość menu - instrukcja warunkowa switch**

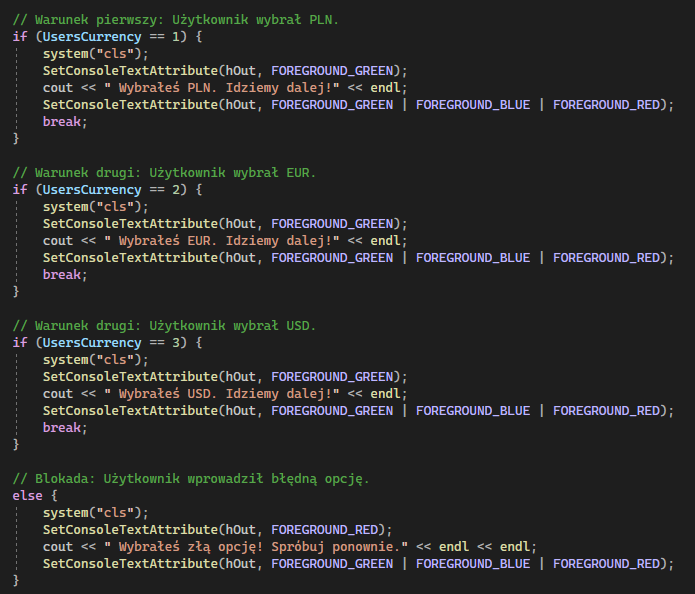
  
Instrukcja warunkowa switch odpowiedzialna za wyświetlanie poszczególnych pozycji menu.

### **Warunek pierwszy**

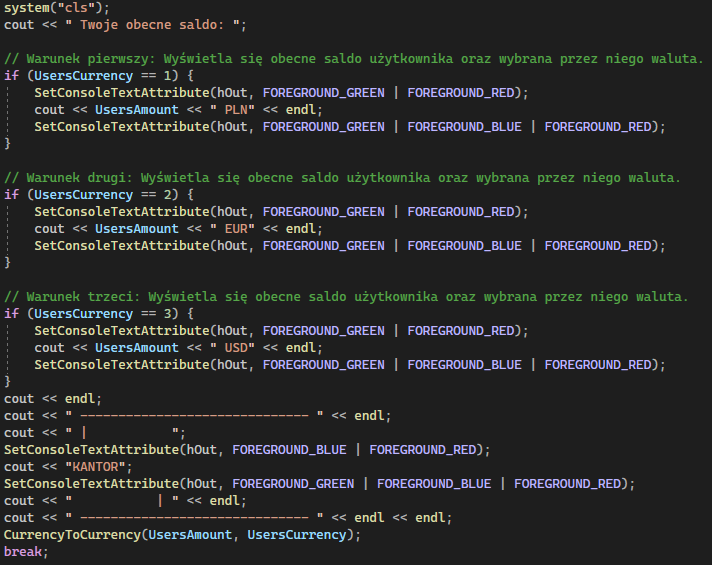
Pierwszy warunek to ponowne wprowadzenie kwoty przez użytkownika. Sprawdza czy wprowadzona kwota zgadza się z podanym zakresem.

  
Jeśli użytkownik wprowadzi liczbę z przedziału od 1 do 999999, to program kontynuuje działanie. W przeciwnym wypadku użytkownik jest proszony o ponowne wprowadzenie kwoty.

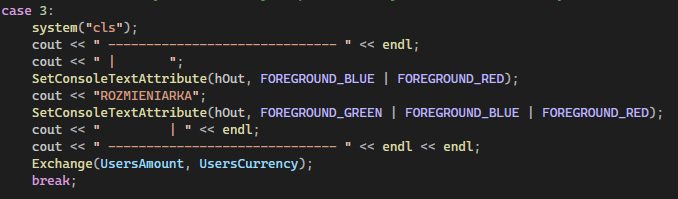
  
Następnie użytkownik ponownie proszony jest o wybranie interesującej go waluty.

  
Sprawdzenie, jaką walutę wybrał użytkownik. Jeśli użytkownik wybrał PLN (wartość 1), zostanie wyświetlony komunikat "Wybrałeś PLN. Idziemy dalej!". Jeśli użytkownik wybrał EUR (wartość 2), zostanie wyświetlony komunikat "Wybrałeś EUR. Idziemy dalej!". Jeśli użytkownik wybrał USD (wartość 3), zostanie wyświetlony komunikat "Wybrałeś USD. Idziemy dalej!". Jeśli użytkownik wprowadził nieprawidłową wartość, na przykład inną niż 1, 2 lub 3, zostanie wyświetlony komunikat "Wybrałeś złą opcję! Spróbuj ponownie." oraz zostanie poproszony o ponowne wprowadzenie odpowiedniej opcji.

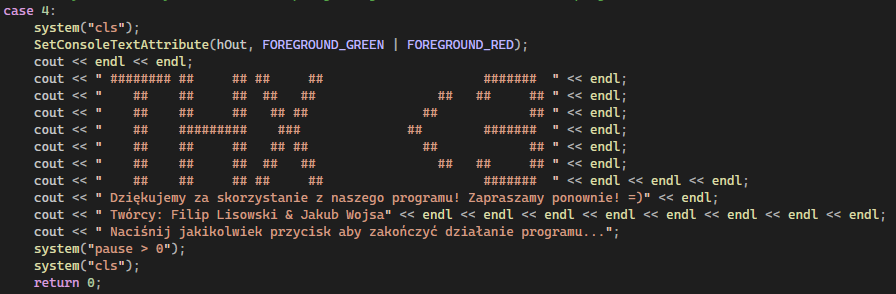
### **Warunek drugi**

  
Warunek drugi odpowiedzialny jest za wywołanie funkcji CurrencyToCurrency(). Jako wartość funkcja przyjmuje UsersAmount i UsersCurrency. Nad wywołaną funkcją wyświetla się aktualne saldo użytkownika oraz napis "KANTOR".

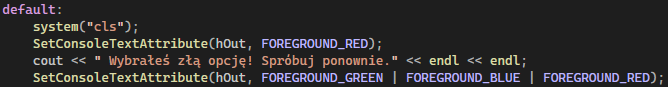
### **Warunek trzeci**

  
Warunek trzeci odpowiedzialny jest za wywołanie funkcji Exchange(). Jako wartość funkcja przyjmuje UsersAmount i UsersCurrency. Nad wywołaną funkcją widnieje napis "ROZMIENIARKA".

### **Warunek czwarty**

  
Warunek czwarty odpowiedzialny jest za wyświetlenie wiadomości pożegnalnej, twórców programu oraz zamknięcie programu.

### **Warunek default’owy**

  
Komunikat o błędnym wprowadzeniu opcji. Prośba o ponowne wybranie opcji.