

Dla programistów C++

Grupa początkująca:

1. Napisz program sumujący kody ASCII danego słowa.
2. Oblicz sumę liczb pierwszych wprowadzanych przez użytkownika.
3. Sprawdź, czy liczba jest wstętna lub czy jest diabelska (Liczby wstętne w zapisie binarnym mają nieparzystą liczbę cyfr 1. W przypadku, gdy liczba nie jest wstętna to mówi się o liczbach Diabelskich – w zapisie binarnym mają parzystą liczbę cyfr 1).

Grupa zaawansowana:

Na zajęciach napisaliśmy program do weryfikacji poprawności numeru PESEL i dlatego teraz proszę o napisanie aplikacji walidującej numer **NIP** lub **IBAN**.

1. NIP

Proszę napisać program do weryfikacji numeru NIP (Numer Identyfikacyjny Płatnika). Jest to numer identyfikujący osoby lub podmioty gospodarcze w systemie rozliczeń podatkowych. Składa się on z dziesięciu cyfr. Pierwsze dziewięć cyfr identyfikuje płatnika, ostatnia jest cyfrą kontrolną. Aby sprawdzić poprawność numeru należy użyć następującego algorytmu:

Pomnożyć kolejne cyfry numeru NIP przez odpowiednie wagi.

Następnie sumujemy wyniki mnożenia.

Wynik sumowania dzielimy modulo przez 11 (modulo – reszta z dzielenia)

Otrzymana cyfra jest sumą kontrolną NIP'u

Przykład:

NIP: 123-456-32-18

Wagi: 6 5 7 2 3 4 5 6 7

NIP: 1 2 3 4 5 6 3 2 1

Wynik: $6 + 10 + 21 + 8 + 15 + 24 + 15 + 12 + 7 = 118$

$118 \bmod 11 = 8$ (cyfra kontrolna 8)

2. IBAN (International Bank Account Number)

Nowe oznakowanie rachunków bankowych jest już prawie zgodne z tym systemem. Dwie początkowe cyferki to właśnie cyfry kontrolne.

65 1060 0076 0000 3200 0005 7153

Aby uzyskać pełną zgodność należy dodać przed numerem litery PL.

PL65 1060 0076 0000 3200 0005 7153

IBAN składa się z dwuliterowego kodu kraju (dla Polski są to litery PL), dwucyfrowej liczby kontrolnej oraz numeru konta bankowego BBAN według systemu wybranego w danym kraju i danym banku. Numer konta BBAN może być dowolnej długości (do 30 znaków) ale długość musi być stała dla danego kraju. Numer konta może zawierać cyfry 0-9 i wielkie litery alfabetu

łacińskiego A-Z. Do celów prezentacyjnych np. na wydrukach można numer IBAN dzielić na grupy 4-ro znakowe oddzielając je znakiem spacji.

Przyjęty przy weryfikacji system cyfr kontrolnych nie posługuje się wagami, ale traktuje numer IBAN jako liczbę całkowitą, która powstaje przez przesunięcie 4 początkowych znaków na koniec (kod kraju i dwucyfrowa liczba kontrolna), zamianę liter łacińskich A-Z na liczby całkowite od 10 do 35.

Weryfikacja składa się z następujących kroków:

1 - usuń niealfanumeryczne znaki - np. spacje, myślniki itd. oraz przesun 4 pierwsze znaki na koniec,

2 - zamień litery na dwucyfrowe liczby; A=10, B=11, ... Y=34, Z=35

3 - podziel całkowicie liczbę z kroku 2 przez 97,

4 - jeśli reszta z dzielenia wyniosła 1 to numer IBAN jest prawidłowy.

Przykład:

1. BE62 5100 0754 7061 -> BE62510007547061
BE62510007547061 -> 510007547061BE62
2. literę B zamieniamy na 11, literę E zamieniamy na 14

510007547061BE62 staje się 510007547061111462

3. Obliczamy resztę z dzielenia;

jeśli reszta =1 to numer jest dobry

510007547061111462 MOD 97 = 1 czyli dobry

W związku z tak dużą liczbą proszę o użycie typu (dla PC i systemu 64 bit): **long long int**

W innym przypadku w trakcie obliczeń należy podzielić długi ciąg cyfr na mniejsze porcje i na nich zrobić obliczenia w trybie tak jak to się robi na papierze. Liczba cyfr w podzielonych liczbach zależy od typu całkowitoliczbowego jaki wybierzemy do reprezentacji tych liczb.

np. 510007547061111462 dzielimy na dwie liczby:

510007547 i 061111462

510007547 MOD 97 = 74

resztę z dzielenia dopisujemy na początku drugiego kawałka

74061111462 MOD 97 = 1 to jest właśnie wynik dzielenia

Dla programistów PHP i MYSQL

Mamy już przygotowane funkcje dodawania, usuwania, wyszukiwania i poprawiania danych w bazie MySQL. Proszę, przygotujcie ładny 😊 interfejs całości programu.

Dla GRAFIKÓW

Doskonalimy podstawowe funkcje w programie GIMP i CorelDraw. Proszę o przygotowanie grafiki na temat epidemii w historii ludzkości.

Dla UCZNIÓW PRACUJACYCH NA UKŁADACH LOGICZNYCH w MMLogic

Program można pobrać ze strony: <http://www.softronix.com/logic.html>

Wykonać tabelę prawdy i zaprojektować układ w MMlogic:

$$y=(a*b)+(c*c)$$

$$y=(a+b+c)*d$$

$$y=(_a*b)+_c$$

(a – sygnał, $_a$ – zaprzeczony sygnał)

Przyślijcie tabelkę prawdy z zrzut ekranu z układem

DRODZY UCZNIOWIE, WYŚLIJCIE WYNIKI PRAC na adres: admin@mdkkielce.pl

Programiści! Rozwiążcie problemy dowolną metodą. Ergonomię kodu przeanalizujemy na zajęciach i wprowadzimy ewentualne poprawki.

Znacie mój nr telefonu, dzwońcie w przypadku problemów!

Trzymajcie się ZDROWO!

Paweł Król