Podstawy Programowania

Ćwiczenia nr 5

wersja 2.1

Temat: Funkcja z argumentami.

Zadanie 1. (1 pkt.)

Działanie programu:
1. Wyświetlenie `Enter 3 numbers:`.
2. Użytkownik wprowadza 3 liczby.
3. Wyświetlenie ich arytmetycznej średniej.

Wymagania: Średnią liczb *a*, *b* i *c* obliczaj i zwracaj funkcją `srednia(a,b,c)`, której argumenty przekazywane są przez wartość.

Zadanie 2. (1 pkt.)

Działanie programu:
1. Wyświetlenie `Enter 1 natural number:`.
2. Użytkownik wprowadza liczbę naturalną.
3. Wyświetlenie silni z podanej liczby.

Wymagania: Silnię liczby *a* obliczaj i zwracaj funkcją `silnia(a)`, której argument przekazywany jest przez referencję.

Zadanie 3. (4 pkt.)

Działanie programu:
1. Wyświetlenie `Enter the maximum value:`.
2. Użytkownik wprowadza liczbę naturalną.
3. Wyświetlenie kolejnych liczb ciągu Fibonacciego które nie przekraczają podanej maksymalnej wartości.

Wymagania: Ciąg Fibonacciego to $n_0 = 0$, $n_1 = 1$, $n_2 = n_0 + n_1$, $\ldots n_i = n_{i-2} + n_{i-1}$.

Element n_i obliczaj i zwracaj funkcją `element(n2,n1)`, której argumenty *n2* = n_{i-2} i *n1* = n_{i-1} przekazywane są przez adres.

Ciąg Fibonacciego od $n_0 = 0$ do $n_i \leq max$ obliczaj i wyświetlaj funkcją `ciag(max)`, której argument przekazywany jest przez adres.

Zadanie 4. (4 pkt.)

Działanie programu:
1. Wyświetlenie `Enter 10 integer numbers:`.
2. Użytkownik wprowadza 10 liczb całkowitych.
3. Program wyświetla `Left or Right?`.
4. Użytkownik wprowadza `Left` (w lewo) lub `Right` (w prawo).
5. Wyświetlenie `Offset:`.
6. Użytkownik wprowadza liczbę.
7. Program przesuwa cyklicznie zawartość tablicy o podaną liczbę komórek w lewo lub prawo, odpowiednio do wyboru użytkownika.
8. Program wyświetla zawartość tablicy.

Wymagania: Wprowadzane liczby zapisuj w kolejnych komórkach statycznej tablicy: liczbę *a* wprowadzaj do *komórki* w *tablicy* funkcją `wprowadz(tablica,komorka,a)`, której argument *tablica* przekazywany jest przez adres, *komorka* – przez referencję i *a* – przez wartość.

Cykliczne przesunięcie zawartości *tablicy* o *liczbę* komórek w wybranym *kierunku* wykonuj funkcją `przesun(tablica,liczba,kierunek)`, której argument *tablica* przekazywany jest przez adres.

Zadanie 5. (1 pkt.)

Działanie programu:
1. Wyświetlenie `Enter a string:`.
2. Użytkownik wprowadza napis o długości ≤ 99 znaków.
3. Wyświetlenie długości tego napisu.

Wymagania: Wprowadzany napis zapisz w statycznej tablicy, pamiętając o znaku końca napisu.
Użyj funkcji `fgets` do wprowadzenia napisu do tablicy.
Długość napisu z *tablicy* wyznaczaj i zwracaj funkcją `dlugosc(tablica)`, której argument przekazywany jest przez adres.

Zadanie 6. (5 pkt.)

Działanie programu:
1. Wyświetlenie `Enter a string:`.
2. Użytkownik wprowadza pierwszy napis o długości ≤ 99 znaków.
3. Wyświetlenie `Enter a string to change from:`.
4. Użytkownik wprowadza drugi napis o długości 5 znaków.
5. Wyświetlenie `Enter a string to change to:`.
6. Użytkownik wprowadza trzeci napis o długości 5 znaków.
7. Zamiana każdego wystąpienia drugiego napisu na trzeci napis w pierwszym napisie.
8. Wyświetlenie pierwszego napisu.

Wymagania: Wprowadzane napisy zapisz w statycznych tablicach, pamiętając o znaku końca napisu.
Użyj funkcji `fgets` do wprowadzenia napisu do tablicy.
Położenie pierwszego znaku drugiego napisu w pierwszym napisie znajduj i zwracaj funkcją `znajdz(napis1,napis2)`.
Zamianę każdego wystąpienia drugiego napisu na trzeci napis w pierwszym napisie wykonuj funkcją `zamien(napis1,napis2,napis3)`.

Zadanie 7. (2 pkt.)

Działanie programu:
1. Wyświetlenie `Enter a string:`.
2. Użytkownik wprowadza napis o długości ≤ 99 znaków.
3. Wyświetlenie `Enter a character:`.
4. Użytkownik wprowadza znak.
5. Zamiana wszystkich spacji na podany znak.
6. Wyświetlenie zmienionego napisu.

Wymagania: Wprowadzany napis zapisz w globalnej statycznej tablicy, pamiętając o znaku końca napisu.
Użyj funkcji `fgets` do wprowadzenia napisu do tablicy.
Napis z *tablicy* zmień funkcją `zmiana(tablica,znak)`, która każdą kolejną literę, jeśli to spacja, zamienia na podany *znak*.