

POLITECHNIKA ŚWIĘTOKRZYSKA
Wydział Elektrotechniki, Automatyki i Informatyki
Katedra Elektrotechniki Przemysłowej i Automatyki
Zakład Urządzeń i Systemów Automatyki

Informatyka 2

Procedury i funkcje

Instrukcja laboratoryjna

Paweł Strączyński

2018

1 Procedury

Procedura służy do zamknięcia wielu instrukcji programu w postaci jednego bloku. Procedury stosuje się zwykle tam gdzie istnieje konieczność wykonywania wielokrotnie tego samego zestawu poleceń. Zamiast wywoływać wiele instrukcji wewnątrz programu można wywołać procedurę (podprogram). Deklaracje procedury w języku Pascal przedstawiono poniżej:

```
procedure nazwa(parametr1 : typ1;...;)
var
    zmienna_lokalna : typ_zmiennej;
    ...
begin
    INSTRUKCJE;
end
```

Deklaracja procedury zaczyna się od słowa kluczowego `procedure` które jest zastrzeżonym słowem kluczowym języka Pascal. Następnie nagłówek procedury zawiera nazwę oraz listę argumentów przekazywanych do jej wnętrza. Procedura (podprogram) może zawierać zmienne lokalne. Zmienne deklarowane są w sposób analogiczny do zmiennych globalnych w programie głównym. Zmienne lokalne widoczne są tylko wewnątrz podprogramu w którym zostały zadeklarowane. Wewnątrz ciała procedury pomiędzy słowami kluczowymi `begin` oraz `end` mogą być wykorzystywane takie same instrukcje jak wewnątrz programu głównego. Wewnątrz ciała procedury mogą zostać użyte zmienne globalne zadeklarowane w programie głównym, zmienne lokalne zadeklarowane wewnątrz procedury oraz przekazywane parametry.

2 Funkcje

Funkcja podobnie jak procedura służy do grupowania sekwencji instrukcji. Od procedury funkcję wyróżnia to, że zwraca ona wartość. Wewnątrz ciała funkcji musi zatem wystąpić co najmniej jedna instrukcja przypisania funkcji zwracanej wartości - zazwyczaj wyniku realizowanych wewnątrz funkcji operacji. Deklarację funkcji przedstawiono poniżej:

```
function nazwa_fun(parametr1 : typ1;...;) : typ_wyniku;
var
    zmienna_lokalna : typ_zmiennej;
```

```

...

begin
    INSTRUKCJE;
    nazwa_fun := wyrażenie;
end

```

Zarówno procedury jak i funkcje muszą być zadeklarowane po deklaracji zmiennych globalnych przed programem głównym. Na listingu poniżej przedstawiono przykład użycia funkcji.

Listing 2.1: Przykład użycia funkcji

```

1 program prog1;
2
3 var spam,ham :Integer;
4
5 function dodaj(foo:Integer; bar:Integer) : Integer;
6 begin
7     dodaj := foo+bar;
8 end;
9
10 begin
11     spam:=2;
12     ham:=5;
13     writeln('Suma= ',dodaj(spam,ham));
14 end.

```

3 Zadania do wykonania

Zadanie 1 Napisać program do obliczania:

- obwodu okręgu o promieniu R ,
- pola powierzchni koła o promieniu R

Każda z operacji powinna być realizowana w postaci odrębnej funkcji. Operacje wczytywania i drukowania danych powinny być zrealizowane w postaci odrębnych procedur.

Zadanie 2 Napisać program wyznaczający pierwiastek kwadratowy N kolejnych liczb naturalnych. Operacja pierwiastkowania powinna być zdefiniowana w postaci odrębnej procedury. Procedura powinna dokonywać stosownych obliczeń dla liczby podanej jako parametr oraz

drukować wynik na ekranie. Program ponadto powinien zawierać funkcję **Potega** która będzie wykorzystywana do wyliczenia pierwiastka.

Zadanie 3 Napisać program weryfikujący poprawność wprowadzonego przez użytkownika numeru PESEL. Program powinien:

- sprawdzić poprawność w oparciu o informacje przedstawione poniżej.

Wskazówka

Numer PESEL jest poprawny jeśli wyrażenie przedstawione poniżej dzieli się bez reszty przez 10..

$$1 \cdot a + 3 \cdot b + 7 \cdot c + 9 \cdot d + 1 \cdot e + 3 \cdot f + 7 \cdot g + 9 \cdot h + 1 \cdot i + 3 \cdot j + 1 \cdot k$$

gdzie litery a-k oznaczają kolejne cyfry numeru. Weryfikacja poprawności numeru PESEL powinna być zrealizowana w postaci odrębnej funkcji.

Zadanie 4 Rozbudować program z zadania poprzedniego o funkcję która dla podanego numeru PESEL (jeżeli jest poprawny) rozpoznawał czy należy do kobiety czy mężczyzny.

Wskazówka

Informacja o płci osoby zawarta jest na 10. cyfrze. Cyfry parzyste oznaczają płć żeńską natomiast nieparzyste męską.