

## Ćwiczenie 2 - Programowanie obiektowe i modułowe 2

1. Opracuj i zapisz w pliku NOTY.PAS moduł zawierający obiektową bibliotekę do wyznaczania oceny stylu skoku narciarskiego. Zastosuj deklarację:

```
type TNoty = object
    procedure UstawDane(a,b,c,d,e: real);
    procedure WprowadzDane; {wczytanie ocen}
    function ZaStyl: real;
private
    X: array[1..5] of real;
    function Max:real; {wyznaczanie oceny najwyższej}
    function Min:real; {wyznaczanie oceny najniższej}
    function Suma:real; {wyznaczanie sumy ocen}
end;
```

Przeznaczenie składowych typu TNoty:

- X – tablica przeznaczona na noty 5 sędziów,
- UstawDane – ustalanie wartości elementów tablicy not na podstawie parametrów,
- WprowadzDane – wprowadzanie not 5 sędziów z klawiatury,
- Max – pomocnicza funkcja zwracająca wartość noty najwyższej,
- Min – pomocnicza funkcja zwracająca wartość noty najniższej,
- Suma – pomocnicza funkcja zwracająca wartość sumy 5 not,
- ZaStyl – funkcja zwracająca obliczoną ocenę stylu skoku równą sumie not pięciu sędziów z odrzuceniem noty najniższej i najwyższej ( $ZaStyl = Suma - Max - Min$ ).

Napisz program testujący moduł.

2. Uzupełnij moduł o deklaracje typu:

```
type TSkok = object
    Noty: TNoty;
    Dl: real;
    procedure UstawDane(a,b,c,d,e,dlugosc: real);
    procedure WprowadzDane;
    function Ocena(Za_m: real; K:integer):real;
end;
```

Przeznaczenie składowych typu TSkok:

- Noty – obiekt typu TNoty do zapisu not 5 sędziów za styl
- Dl – dla zapisu długości skoku,
- UstawDane – ustalanie wartości składowych (not 5 sędziów i długości skoku) wg parametrów,
- WprowadzDane - wprowadzanie danych - długości skoku i 5 not sędziów – z klawiatury,
- Ocena - wyznaczenie oceny skoku równej sumie oceny za styl (metoda ZaStyl obiektu Noty) i oceny za długość. Ocena za długość wyznaczana jest z wykorzystaniem parametrów metody:  
K – odległość kalkulacyjna (punkt kalkulacyjny) skoczni (dla Wielkiej Krokwi – 120)

Za\_m - liczba punktów przyznawana za każdy metr długości skoku przekraczającej punkt k (dla Wielkiej Krokwi – 1.8)

wg wzoru:

$$\text{Ocena} = 60 + (\text{Dl} - \text{K}) * \text{Za}_m$$

Zapisz zmieniony moduł w pliku SKOK.PAS. Napisz program testujący zmieniony moduł.

### 3. Uzupełnij moduł o deklaracje typu:

```
type TSkoczek = object
  Nazwisko, Kraj: string[20];
  Skok1, Skok2: TSkok;
  procedure UstawNazw(n,k:string);
  procedure WyswWyniki(za_m:real; k:integer);
end;
```

Moduł przeznaczony jest do zapisu i przetwarzania danych o dwu skokach wykonanych przez zawodnika w konkursie skoków narciarskich.

Przeznaczenie składowych typu TSkok:

Nazwisko, Kraj – do zapisu imienia, nazwiska oraz kraju reprezentowanego przez zawodnika,

Skok1, Skok2 – obiekty typu TSkok do zapisu danych o dwu skokach oddanych przez zawodnika w konkursie,

UstawNazw – ustalenie nazwiska i kraju na podstawie parametrów.

WyswWyniki – wyświetlanie jednowierszowej informacji o wynikach skoczka zawierającej nazwisko, kraj, długości dwu skoków i dwie noty za skoki.

Zapisz zmieniony moduł w pliku SKOCZEK.PAS. Napisz program testujący moduł.

Zadanie domowe:

### 4. Opracuj i zapisz w pliku ZAWODY.PAS moduł zawierający deklarację typu obiektowego:

```
type TZawody = object
  miasto, kraj: string;
  za_m: real;
  k, n: integer;
  x: array[1..50] of TSkoczek;
  procedure WprowadzZPliku(nazwa:string);
  procedure WyswWyniki;
end;
```

Algorytmy realizowane przez poszczególne metody:

WprowadzZPliku – organizacja wprowadzania danych o konkursie skoków z pliku tekstowego o nazwie określonej parametrem.

WyswWyniki – wyświetlanie tabeli rezultatów konkursu.

Napisz program testujący moduł.