

Algorytmy i struktury danych

Typ wskaźnikowy
Zmienne dynamiczne

Witold Marańda
maranda@dmcs.p.lodz.pl

1

Zmienne statyczne

Omawiane do tej pory typy danych, tj. proste (wbudowane) i strukturalne pozwalają na deklarowanie i używanie w programie zmiennych statycznych. Mają one następujące cechy:

- ◆ Użycie ich musi być przewidziane przez programistę, gdyż tworzone są w sekcji deklaracji zmiennych na początku programu (podprogramu) i istnieją do końca wykonywania programu (podprogramu),
- ◆ Zajmują w pamięci operacyjnej komputera z góry ustalony rozmiar, który nie może być zmieniany w czasie działania programu (np. rozmiar tablicy, raz ustalony nie może być zmieniony)

2

Zmienne dynamiczne

Ograniczeń tych pozbawione są zmienne dynamiczne, które mają następujące cechy:

- ◆ nie jest konieczne przewidywanie liczby i rozmiaru zmiennych dynamicznych, gdyż powoływane i usuwane są na żądanie programisty, w dowolnym miejscu programu,
- ◆ zmienne dynamiczne zajmują taki obszar pamięci jaki jest w danej chwili wymagany dla przechowania danych, a po użyciu obszar ten może być zwolniony do innego użycia.

Zmienne dynamiczne reprezentują obiekty, dla których pamięć jest przydzielana i zwalniana na żądanie programisty.

3

Typ wskaźnikowy

Zmienne dynamiczne nie posiadają identyfikatorów (nazw), a odwołanie do nich odbywa się za pomocą pomocniczych zmiennych zwanych wskaźnikami.

Wskaźniki są obiektami specjalnego typu danych, tj. typu wskaźnikowego. Wartościami obiektów typu wskaźnikowego są położenia w pamięci (adresy) zmiennych dynamicznych.

Zmienne wskaźnikowe mogą być zmiennymi statycznymi lub dynamicznymi.

Deklaracja:

$$\text{type } T = \wedge T_0$$

gdzie T jest nazwą typu, a T_0 jest nazwą typu wskazywanego.

W deklaracji typu wskaźnikowego występuje znak '^' umieszczony przed nazwą typu wskazywanego.

4

Typ wskaźnikowy

Przykład deklaracji typu (PASCAL):

```
TYPE wr = ^real;
TYPE wi = ^integer;
TYPE wrec = ^RECORD
    ...
END
```

Przykład deklaracji zmiennych (PASCAL):

```
VAR
    x : wr;
    y : ^real;
    i, j, k : ^integer;
```

Na zmiennych typu wskaźnikowego (wskaźnikach) możliwe są następujące operacje:

- przypisanie wartości jednego wskaźnika innemu wskaźnikowi, np.

$x := y$

- relacja, tj. badanie czy wskaźniki mają identyczne wartości, co oznacza, że wskazują na ten sam obiekt w pamięci, np.

$x = y$ lub $x <> y$

(x i y muszą być zmiennymi wskaźnikowymi tego samego typu)

5

Typ wskaźnikowy

Odwołanie do wartości obiektu wskazywanego odbywa się poprzez operację wyłuskania. Wyłuskanie oznacza się znakiem '^' umieszczonym za nazwą zmiennej dynamicznej. Tak utworzona nazwa (tj. nazwa^) jest identyfikatorem zmiennej dynamicznej i może być używany w normalny sposób, np.:

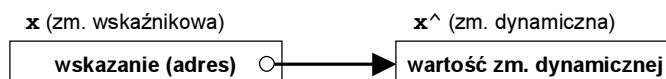
$k := x^;$	przypisanie wartości zmiennej dynamicznej do zmiennej statycznej
$y^ := x^;$	przypisanie wartości zmiennej dynamicznej do zmiennej dynamicznej
$x^ < k;$	porównanie zmiennej dynamicznej i zmiennej statycznej

(x i y muszą być zmiennymi wskaźnikowymi tego samego typu, a k zmienną typu zgodnego ze wskazywanym przez x i y)

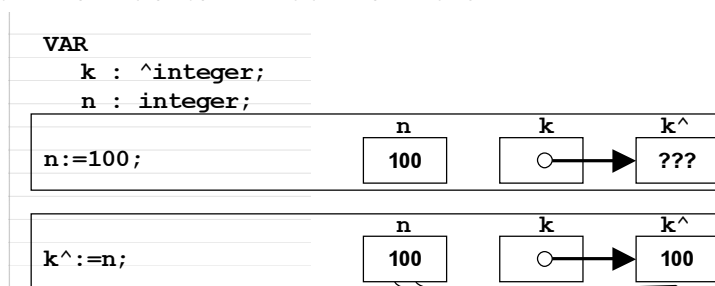
6

Typ wskaźnikowy – interpretacja graficzna

Stosowana jest następująca interpretacja graficzna dla wskaźników i zmiennych dynamicznych



Przykład operacji przypisania i jej interpretacja graficzna:



7

Tworzenie i usuwanie zmiennych dynamicznych

Tworzenie i usuwanie zmiennych odbywa się wewnątrz programu za pomocą procedur **New()** i **Dispose()**.

Procedury te wymagają jako argumentu zmiennej wskaźnikowej. W podanych poniżej przykładach zmienna p jest zmienną wskaźnikową do obiektów typu T_0 .

Procedura NEW(p) – tworzenie zmiennych dynamicznych typu T_0 :

1. utworzona zostaje zmienna dynamiczna typu T_0 o wartości nieokreślonej (rezerwacja pamięci);
2. zmiennej p zostaje przypisana nowa wartość będąca wskazaniem (adresem) nowo utworzonej zmiennej.

Do wartości tak utworzonej zmiennej dynamicznej można się odwoływać poprzez nazwę: $p^$.

Procedura DISPOSE(p) – usuwanie zmiennych dynamicznych:

1. zmiennej p zostaje przypisana wartość nieokreślona, tj. zmienna p nie wskazuje już na żaden określony obiekt w pamięci;
2. zwolniona zostaje pamięć zajmowana przez zmienną dynamiczną i może być od tej pory wykorzystana do innych celów.

8

Zmienne dynamiczne - przykład

Przykład fragmentu programu i jego interpretacja graficzna:

