

### 30. Príkaz case v jazyku Pascal, funkcie readkey a keypressed.

Vetvenie CASE rozdeľuje program na viac alternatívnych možností riešenia problému na základe dosiahnutia určitej hodnoty výberovej premennej (selektora). Tam, kde potrebujeme rozlíšiť viac hodnôt jednej premennej, môžeme namiesto viacerých vnorených IF.. použiť príkaz CASE..., ktorý obsahuje zoznam významných hodnôt a k nim priradené príkazy alebo programové bloky

Neúplný tvar:

```
case premenná of
    hodnota1 : príkaz1;
    hodnota2: príkaz2;
    ...
    hodnotan: príkazn;
end;
```

Úplný tvar:

```
case premenná of
    hodnota1 : príkaz1;
    hodnota2: príkaz2;
    ...
    hodnotan: príkazn;
else príkazn+1;
end;
```

#### Fungovanie:

Podľa toho, akú hodnotu má premenná, vykoná sa príslušný príkaz a pokračuje sa za príkazom case. V prípade, že hodnota premennej sa nerovná hodnote<sub>1</sub> až hodnote<sub>n</sub>, vykoná sa príkaz za kľúčovým slovom else (alebo sa nevykoná nič).

case sa oplatí použiť ak chceme vetvenie s viacerými možnosťami ako 2.

Premenná môže byť ľubovoľného ordinálneho typu + string, zväčša jednoduchá premenná. Jednotlivé vetvenia majú vľavo od dvojbodky zoznam konštánt, pre ktoré toto vetvenie platí (okrem jednotlivých hodnôt sa v ňom môžu vyskytnúť aj intervaly).

Prvky sú od seba oddelené čiarkami a za dvojbodkou nasleduje príkaz, ktorý sa vykoná keď premenná nadobúda hodnotu uvedenú vľavo.

postup: najprv sa vyhodnotí premenná, nájde sa vetva vyhovujúca výsledku premennej potom sa vykoná príslušný príkaz.

Poznámka:

Typ: byte, integer

1  
1..3  
1,2,3

Char (jeden znak ASCII) prip. string

‘a’  
‘+’  
‘a’, ‘b’  
‘auto’

V prípade potreby vykonania viacerých príkazov v rámci napr. príkazu1 použijeme *begin* a *end*.

## Príklad na príkaz CASE

Napište program, v ktorom zistujete, či znak je veľké, malé písmeno alebo číslica.

```
program znaky;
uses crt;
var q:char;
begin
  clrscr;
  repeat
    write('Zadaj znak z ASCII tabulky');
    readln(q);
    if ((q<>'k') or (q<>'K')) then
      case q of
        'A'..'Z': writeln(q,' je velke pismeno. ');
        'a'..'z': writeln(q,' je male pismeno. ');
        '0'..'9': writeln(q,' je cislica. ');
      else
        writeln(q,' nie je ani pismeno, ani cislica. ');
      end;
    until ((q='k') or (q='K'));
  readln;
end.
```

### KeyPressed: boolean;

Logická funkcia, ktorá vráti TRUE ak je niečo v buffri klávesnice(bolo niečo stlačené) a vráti FALSE, ak je buffer klávesnice prázdny(ak nebolo nič stlačené). Takto môžeme napr. zobrať vstup z klávesnice len ak bolo niečo stlačené, a ak nebolo nič stlačené, program môže bez zastavenia pokračovať ďalej. Deklarácia je pomocou typu premennej boolean.

```
program vypisx;
uses crt;
begin
  writeln('x');
  repeat until keypressed;
end.
```

### ReadKey: char;

Funkcia, ktorá vráti znak stlačenej klávesy. Ak je buffer klávesnice prázdny- program sa zastaví a čaká na stlačenie znaku na klávesnici. Readkey oproti vstupu pomocou read sa líši v tom: Readkey nevyžaduje stlačiť enter, nezmení obrazovku(stlačený kláves sa nezobrazuje) a nemení polohu kurzora na obrazovke. Deklarácia je pomocou typu premennej char.

```
program klaves;
uses crt;
var kl:string;

begin
  clrscr;
  repeat
    writeln ('Stlac klaves (Q,q: koniec) ');
    kl:=readkey;
    writeln(kl);
  until (kl='q') or (kl='Q');
end.
```