

PASCAL – INSTRUKCJE WARUNKOWE

- **Zadanie 1:** Przypominając sobie wiadomości z poprzedniej sekcji, napisz program obliczający różnicę pomiędzy wartościami kodów dowolnej (podanej przez użytkownika) dużej litery oraz jej małego odpowiednika. Sprawdź wynik dla różnych liter, czy dostrzegasz prawidłowość?

1. Instrukcja warunkowa

- Zapewnia możliwość zmiany przebiegu programu w zależności od zadanego warunku.
- **JEŻELI** warunek jest spełniony, **TO** wykonaj operację
- **W PRZECIWNYM RAZIE** wykonaj inną operację
- W Pascalu powyższe zdanie zapisać należy tak:

```
IF warunek THEN
  {blok instrukcji}
ELSE
  {blok instrukcji}
```
- Można pominąć klauzulę **ELSE**, a wtedy program po prostu pójdzie dalej jeśli warunek nie będzie spełniony
- Napisz program, który pyta użytkownika o wynik działania $2*2$, jeśli użytkownik poda właściwy wynik, pochwali go, a w przeciwnym razie skarci.

```
program pierwszy_if;
  uses crt;
  var
    s: char;
begin
  clrscr;
  write ('Podaj wynik iloczynu 2*2: ');
  s:=readkey;
  if (s='4') then {s='4' to warunek logiczny, który zwraca wartość TRUE lub FALSE}
  begin
    writeln (' Brawo! Bardzo dobrze! Jestem na prawdę dumny!');
  end
  else           {zwróć uwagę, że przed else nie występuje średnik}
    writeln (' No nie za dobrze, słabo, źle. Spróbuj innym razem');
  readln;
end.
```

2. Pierwiastki równania kwadratowego

- Instrukcje warunkowe można zagnieżdżać.
- Mamy dane równanie kwadratowe:
 - $ax^2 + bx + c = 0$,
 - jeśli $a=0$, to nie jest równanie kwadratowe,
 - należy najpierw policzyć wyróżnik równania kwadratowego
 - $\Delta = b^2 - 4ac$
 - w zależności od niego:
 - jeżeli $\Delta > 0$, to równanie ma dwa rozwiązania rzeczywiste (dwa różne pierwiastki rzeczywiste), są to:
 - $x_1 = \frac{-b - \sqrt{\Delta}}{2a}$ oraz $x_2 = \frac{-b + \sqrt{\Delta}}{2a}$
 - jeżeli $\Delta = 0$, to równanie ma jedno rozwiązanie rzeczywiste (jeden podwójny pierwiastek rzeczywisty);
 - $x_0 = \frac{-b}{2a}$
 - jeżeli $\Delta < 0$, to równanie nie ma rozwiązań rzeczywistych,
 - Program według tego sposobu działania będzie wyglądał następująco:

```
program rownanie_kwadratowe;
var
  a,b,c,delta,x1,x2 : real;
begin
  writeln ('Podaj współczynniki równania kwadratowego: ');
  writeln ('Podaj A: '); readln(a);
  if (a=0) then {to nie jest równanie kwadratowe}
    writeln ('To nie jest równanie kwadratowe!')
  else
    begin
      writeln ('Podaj B: '); readln(b);
      writeln ('Podaj C: '); readln(c);
      delta:=sqr(b)-4*a*c;
      if (delta < 0) then {wyróżnik ujemny - nie ma pierwiastków}
        writeln ('Równanie nie ma pierwiastków rzeczywistych')
      else
        if (delta=0) then {wyróżnik zerowy - dokładnie jeden pierwiastek}
          begin
            x1:=-b/(2*a);
            writeln ('Równanie ma jeden pierwiastek rzeczywisty: ', x1:10:5);
          end
        else
          begin
            x1:=(-b-sqrt(delta))/(2*a);
            x2:=(-b+sqrt(delta))/(2*a);
            writeln ('Równanie ma pierwiastki: ', x1:10:5, ' ', x2:10:5);
          end;
        end;
      end;
    readln;
  end.
```

3. Instrukcja wyboru – `case`

- Jeśli jest wiele wariantów wykonywania instrukcji w zależności od wartości jakiejś zmiennej, to możemy się w tym celu posłużyć instrukcją wyboru `case`. Przedstawię najpierw przykład jej użycia, a następnie wytłumaczę:

```
program uzycie_case;
var
  liczba: byte;
begin
  writeln('Podaj jakąś liczbę od 1 do 3:');
  readln(liczba);
  case liczba of
    1: writeln('Podales jeden');
    2: writeln('Podales dwa');
    3: writeln('Podales trzy');
  end;
  readln;
end.
```

- Między słowami `case` i `of` charakterystycznymi dla tej instrukcji podajemy jakąś zmienną całkowitą (oprócz zmiennej typu `longint`), czyli np. zmienną znakową. Nie może to być natomiast zmienna rzeczywista! W następnym linijkach piszemy wartości, po nich dwukropek i instrukcję, która ma być wykonana, jeśli zmienna ma daną wartość. Całą instrukcję kończymy słowem `end`. Inny przykład z użyciem tej instrukcji.

```
program uzycie_case2;
uses crt;
var
  c: char;
begin
  writeln('Wcisnij litere t lub n');
  c:=readkey;
  case c of
    't':
      begin
        writeln('Wcisnales t');
        readln;
      end;
    'n':
      begin
        writeln('Wcisnales n');
        readln;
      end;
  end;
end;
end.
```

- Porównanie instrukcji `case` i `if`:
Instrukcja `case` jest ograniczona typem danych, jakie można porównywać - dozwolone są tylko typy porządkowe - odmiennie do instrukcji `if`. W wielu jednak przypadkach te dwie instrukcje można stosować zamiennie.

4. Zadanie 2

- Napisz program pytający użytkownika, czy chce wykonać działanie $2*(2+2)$. Jeśli użytkownik wciśnie literę `T` (lub `t`), obliczy i poda wynik, jeśli literę `N` (lub `n`), wydrukuje na ekranie „Nie, to nie!” i dopiero zakończy po wciśnięciu `Enter`. Jeśli jakikolwiek inny klawisz, otrzyma komunikat „Wcisnąłeś inny klawisz, za karę poczekasz 10 sekund w bezruchu”
- Jeśli skończyłeś, zawołaj mnie i poproś o punkt.