

LEKCJA 08 – testowanie klawiatury

Potrafiemy już przemieszczać obiekty po ekranie monitora. Co prawda są to kwadraty, ale nic nie stoi na przeszkodzie, aby napisać odpowiednie procedury i zamienić kwadraty na „statki kosmiczne”. Potrafiemy sprawdzić, kiedy znajdują się w odpowiednim punkcie ekranu (zderzą się z innym obiektem). Jeśli dołożymy jeszcze umiejętność reagowania na wciskane klawisze, będziemy mogli pisać prawdziwe gry komputerowe!

Informacja o wciskanych klawiszach przechowywana jest w tzw. „buforze klawiatury”. System pobiera je kolejno i wykonuje polecenia. Jeśli zbyt „znęcałeś” się nad klawiaturą, być może zdarzyło się, że usłyszałeś charakterystyczne piski. To znak, że bufor klawiatury się przepełnił - program nie jest w stanie tak szybko wykonać kolejnych operacji.

Poniższy prosty program służy do sprawdzania kodów wciskanych klawiszy. Zdarzenie ONKEYDOWN formatki.

```
procedure TForm1.FormKeyDown  
(Sender: TObject; var Key: Word; Shift: TShiftState);  
begin  
  ShowMessage(IntToStr(key));  
end;
```

PRZYKŁAD LEKCJA081 - PRZESUWANIE KWADRATU

Poniższy program będzie przesuwał kwadrat zgodnie z kierunkiem naciskanych kursorów. Koniec programu nastąpi po naciśnięciu klawisza ESC. Zmienna KONIEC zawiera początkowo wartość FAŁSZ, Gdy w trakcie działania programu (po naciśnięciu klawisza ESC) przyjmie wartość PRAWDA pętla zakończy swoje działanie

- zadeklaruj dwie zmienne globalne
koniec:boolean=false;
klawisz:word;
- w procedurze do zdarzenia ONKEYDOWN formatki wpisz:
klawisz:=key
- w procedurze do zdarzenia ONRESIZE formatki wpisz program z ramki

W pętli REPEAT animowany jest kwadrat. Kwadrat będzie przesuwany o 5 pikseli w zależności od zawartości zmiennej KLAWISZ. Sprawdzanie klawiatury następuje w momencie uruchomienia procedury **application.ProcessMessages;** która za pomocą procedury **FormKeyDown** testuje klawiaturę i wciśnięty klawisz przypisuje do zmiennej KLAWISZ. Klawisz o numerze 39, to kursor „w prawo” – zwiększenie współrzędnej X o 4 itd. Na zakończenie zmienna KLAWISZ jest zerowana, żeby kolejne przesunięcie odbyło się po naciśnięciu klawisza jeszcze raz. Po naciśnięciu klawisza ESC (kod 27) pętla kończy działanie i formatka jest automatycznie zamykana (CLOSE).

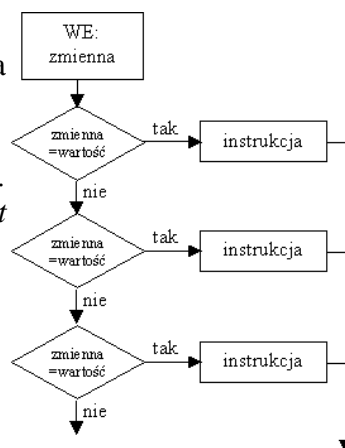
```
var x,y,bok:integer;  
begin  
  x:=0;  
  y:=0;  
  bok:=100;  
  REPEAT  
    kwadrat(x,y,bok,clRed);  
    sleep(10);  
    kwadrat(x,y,bok,clBtnFace);  
    application.ProcessMessages;  
    case klawisz of  
      39: x:=x+5;  
      37: x:=x-5;  
      40: y:=y+5;  
      38: y:=y-5;  
      27: koniec:=true;  
    end;  
    klawisz:=0;  
  UNTIL koniec;  
  close;  
end;
```

Instrukcja wyboru

W programie zastosowano **instrukcję wyboru** - Zamiast wielokrotnego sprawdzania za pomocą warunków logicznych IF THEN.

```
CASE zmienna OF  
  wartość1: instrukcja;  
  wartość2: instrukcja;  
  ...  
  ELSE instrukcja  
END;
```

Porównujemy zmienną z kolejnymi wartościami. Jeśli jest taka sama, to wykonywana jest instrukcja przy tej wartości.



PRZYKŁAD LEKCJA082 - Zmiana wymiarów i kolorów

Kwadratową piłkę umiemy przesuwac po ekranie. Dołożymy zmianę wymiarów piłki i kolorów. Na przykład po naciśnięciu klawisza S (mała litera s – kod 83) piłka będzie zwiększać bok. Odwrotne działanie będzie miał klawisz A (kod 65) - skraca bok. Klawisz z literą K (kod 75) będzie losowo wybierał kolor - należy w procedurze zadeklarować zmienną KOLOR.

PRZYKŁAD LEKCJA083 – PING-PONG

Wykorzystujemy program, w którym kwadratowa piłka odbija się od brzegów ekranu (LEKCJA071)

- dopisać procedurę RAKIETKA - prostokąt symulujący raketkę tenisową.
- zadeklarować zmienne RX i RY – położenie raketki na ekranie
- ustawić początkowe wartości: Rx=600, Ry=200
- raketka przesuwana będzie kursorami w górę i w dół – kody 40 i 38
- raketka ma szerokość 5, wysokość 120 i kolor czarny.
- raketka nie może wyjechać poza ekran w górę i w dół.
- po odbiciu od prawego brzegu nie zmieniamy kierunku (jak dotychczas), lecz kończymy program.
- odbicie nastąpi, gdy piłka odbije się od raketki (analiza poniżej)

Jak sprawdzić czy piłka opuści planszę z prawej strony? Wystarczy, że współrzędna X piłki będzie większa niż 640. A na osi Y? Najbardziej skrajne położenia piłki i raketki opisuje rysunek.

Skrajne położenia raketki:

lewy górny róg R1 - (600,Ry)

lewy dolny róg R2 - (600,Ry+120)

Skrajne położenie piłki:

prawy dolny róg P1 - (x+bok,y+bok)

prawy górny róg P2 - (x+bok,y)

Będziemy sprawdzać tylko te skrajne położenia, a wszystkie pośrednie będą oznaczały, że piłka się odbije i muszą być spełnione następujące warunki:

I - prawy brzeg piłki musi być równy 600, bezpieczniej większe niż lub równe $x+bok \geq 600$

II - dół piłki musi być większy niż góra raketki (róg nas nie interesuje, dlatego $y+bok > Ry$)

III - góra piłki musi być mniejsza niż dół raketki (róg nas nie interesuje, dlatego $y < Ry+120$)

Jednocześnie muszą zachodzić warunki I i III oraz I i II żeby nastąpiło odbicie, co zapiszemy

PRZYKŁAD – LEKCJA084 – ping-pong z timerem

- ustawienia początkowe: x:=0; y:=0 itd. wstawić do procedury od zdarzenia ONCREATE formatki
- wstawić TIMER
- ustawić INTERVAL na 10
- procedura do zdarzenia ONTIMER (przekopiować z LEKCJA083 i zmienić kolejność:
 - 1) usuwanie,
 - 2) nowe współrzędne i sprawdzanie,
 - 3) rysowanie

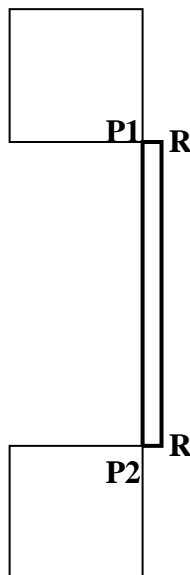
```
kwadrat(x,y,bok,kolor);
sleep(10);
kwadrat(x,y,bok,clBtnFace);
application.ProcessMessages;
case klawisz of
  39: x:=x+5;
  37: x:=x-5;
  40: y:=y+5;
  38: y:=y-5;
  83: bok:=bok+5;
  65: bok:=bok-5;
  75: kolor:=random(maxint);
  27: koniec:=true;
end;
```

```
repeat
  kwadrat(x,y,bok,kolor);
  raketka(Rx,Ry,5,120,clBlack);
  sleep(10);
  kwadrat(x,y,bok,clBtnFace);
  raketka(Rx,Ry,5,120,clBtnFace);
  application.ProcessMessages;
  x:=x+dx;
  y:=y+dy;

  if x < 0 then dx:=-dx;
  if x+bok > 640 then koniec:=true;
  if (y+bok > 480) or (y < 0) then dy:=-dy;

  if (x+bok >= 600) and
    ((y+bok > Ry) and (y < Ry+120))
    then dx:=-dx;

  case klawisz of
    40: if Ry+120<480 then Ry:=Ry+5;
    38: if Ry>0 then Ry:=Ry-5;
    27: koniec:=true;
  end;
  klawisz:=0;
until KONIEC;
```



- wymazywanie rakiетки i piłki
- nowe współrzędne piłki
- sprawdzanie odbić od brzegów piłki
- sprawdzanie odbicia od rakiетки
- czy piłka wyleciała za ekran
- reakcja na wciśnięte klawisze
- rakiетка w dół
- rakiетка w górę
- ESC – koniec
- jeśli tego nie będzie, rakiетка będzie sama się przesuwać
- rysowanie rakiетки i piłki

```

kwadrat(x,y,bok,clBtnFace);
rakiетка(Rx,Ry,5,120,clBtnFace);
x:=x+dx;
y:=y+dy;
if x < 0 then dx:=-dx;
if (y+bok > 480) or (y < 0) then dy:=-dy;
if (x+bok >= 600) and
((y+bok>Ry) and
(y<Ry+120)) then dx:=-dx;
if x+bok > 640 then
begin
showmessage('KONIEC');
close;
end;
case klawisz of
40: if Ry+120<480 then Ry:=Ry+5;
38: if Ry>0 then Ry:=Ry-5;
27: begin
showmessage('KONIEC');
close;
end;
end;
klawisz:=0;
kwadrat(x,y,bok,kolor);
rakiетка(Rx,Ry,5,120,clBlack);

```

Ćwiczenia do programu - LEKCJA082

- po wciśnięciu 'k' obiekt zmienia swój kolor, cyklicznie od 1 do 15,... z 15 na 1 (w koło)
- po wciśnięciu 'l' obiekt zmienia swój kolor, cyklicznie, ale wstecz, z 15 na 14, ... z 1 na 15, w koło
- po wciśnięciu 'A' obiekt przesuwa się (płynna animacja) do lewego brzegu ekranu
- 'S' - prawego brzegu
- 'Z' - dolnego brzegu
- 'W' - górnego brzegu

Ćwiczenia do programu LEKCJA083

- po każdym odbiciu prędkość piłki (zmienne DX i DY) zwiększa się o 1
- po odbiciu od górnego i dolnego brzegu zwiększa się DY
- po odbiciu od lewego i prawego brzegu zwiększa się DX
- po każdym odbiciu zmienia się losowo kolor piłki
- po 10 odbiciach od rakiетки program sam kończy swoje działanie

PRZYKŁAD – LEKCJA085 –

Do sprawdzania reakcji na klawiaturę i myszkę służą w Delphi zdarzenia ONKEY... i ONMOUSE...

- umieść na formacie trzy przyciski
- kolejnym przyciskom przypisz jedno ze zdarzeń związanych z myszką: OnMouseDown, OnMouseMove, OnMouseUp
- z jedną instrukcją: SHOWMESSAGE('ONMOUSE...');
- wypróbuj działanie myszki na przyciskach

W podobny sposób wypróbuj działanie zdarzeń OnKeyDown, OnKeyPress, OnKeyUp. Również trzy przyciski. Zwróć jednak uwagę, że reakcja na klawiaturę nastąpi tylko wtedy, gdy odpowiedni przycisk jest w tzw. **FOKUSIE** – jest wybrany myszką lub klawiaturą, np. kursory lub TAB.