

```

program azdlug;
{Obliczenie azymutu i dlugosci ze wspolrzednych - wersja szkolna - Turbo Pascal}

uses
  CRT, dos;           {deklaracja modulu CRT i DOS przy pomocy USES}

var
  koniec, iw: integer; {zmienne typu integer - calkowite}
  rok,mc,dz,dzt: word; {rok,m_c,dzien,dzien_tyg}

{----- Procedury i funkcje -----}

procedure czekaj; {Czeka na baciwienie Enter i czysci ekran}
var i: integer;   {Deklaracja zmiennej i typu integer}

begin
  {Poczatek procedury}
  writeln;
  write('Nacisnij <Enter> '); {Pisze na ekranie }
  readln;                      {Czeka na nacisnienie klawisza}
  clrscr;                       {Czysci ekran}
end;                          {Koniec procedury}

{Funkcja }
FUNCTION dlug (dx: real; dy:real): real; {Funkcja o 2 parametrach: dx, dy typu
real i zwraca wynik real}

var d: real;           {Zmienna dodatkowa w funkcji typu real}

begin

  d := dx * dx + dy * dy;   {Kwadrat dlugosci}
  d := SQRT(d);             {Pierwiastek z kwadratu dlugosci czyli dlugosc}
  dlug := d;                { W funkcji musi byc podstawienie pod nazwe
funkcji}

end;

{Procedury Dlfun}
Procedure DlFun; {Dlugosc z funkcji}

var x1,y1, x2, y2, dx, dy, d1: real;

Begin
  clrscr;
  writeln;
  writeln( 'Obliczenie dlugosci ze wspolrzednych (z funkcji)');
  writeln;
  write('x1 = ');
  readln(x1);           {Czytanie wspolrzednej x1 punktu pierwszego}
  write('y1 = ');      {Czytanie wspolrzednej y1 punktu pierwszego}
  readln(y1);
  write('x2 = ');      {Czytanie wspolrzednej x2 punktu drugiego}
  readln(x2);
  write('y2 = ');      {Czytanie wspolrzednej y2 punktu drugiego}
  readln(y2);

  dx := x2 - x1;       {Obliczenie dx}
  dy := y2 - y1;       {Obliczenie dy}

  writeln('dx = ',dx:10:3,' dy = ',dy:10:3);
  d1 := dlug(dx, dy); {Dlugosc z funkcji dlug(dx,dy)}
  writeln;
  writeln('D = ', d1:10:3); {Wydruk dlug z dokl 10 przed kropka (ew spacje z
przodu) i 3 miejsca po kropce}

  czekaj; {Wywołanie procedury czekaj}

```

end;

{Procedura Dlugosci}

Procedure Dlugosci;

var x1,y1, x2, y2, dx, dy, d1: real;
i: integer;

Begin

clrscr;

for i:=1 **to** 3 **do** writeln; *{3 puste linie}*

writeln('Obliczenie dlugosci ze wspolrzednych ');

write ('x1: ');

readln(x1);

write('y1: ');

readln(y1);

write('x2: ');

readln(x2);

write('y2: ');

readln(y2);

dx := x2 - x1;

dy := y2 - y1;

d1 := SQRT(dx * dx + dy * dy);

writeln;

writeln('D = ', d1:10:3);

czekaj;

end;

{Procedura Azymut}

procedure azymut;

var

rg, rs, x1, y1, x2, y2, dx, dy, d, a, azg, azs, pi1: real;

i: integer;

begin

pi1 := 4.0 * arctan(1.0); *{Pi obliczone, Pi jest w systemi}*

rg := 200.0 / pi; *{Ro gradowe}*

rs := 180.0 / pi; *{Ro stopniowe}*

clrscr; *{Kasowanie ekranu}*

for i:=1 **to** 3 **do** writeln; *{3 razy pusta linia na ekranie}*

writeln('Program oblicza dlugosc i azymut ');

writeln; *{pusta linia}*

writeln('Pi = ', pi, ' Pi_obl = ', pi1);

writeln;

writeln('Podaj x1 y1 x2 y2 (oddzielone spacja): ');

readln(x1, y1, x2, y2); *{Czytanie wspolrzednych 2 punktów}*

dx := x2-x1; *{ dx }*

dy := y2-y1; *{ dy }*

d :=dlug(dx, dy); *{dlugosc}*

writeln;

writeln('Wyniki: ');

writeln;

writeln('dx= ', dx:10:3, ' dy= ', dy:10:3, ' d=',d:10:3);

if dx = 0 **then** *{Badanie czy dx =0}*

```

begin
  if (dy>0) then           {Jesli dx=0 i dy>0 to }
    a := pi / 2
  else                     {Jesli dx=0 i dy<=0 to }
    a := 1.5 * pi;
  end
  else                     {Jesli dx <> 0}
  begin
    a := arctan(dy / dx);   { obliczenie azymutu w radianach}
    if (dx < 0) then       { Jesli dx<0 }
      a := a + pi
    else                   { Jesli dx >0}
      begin
        if (dy < 0) then   {Jesli dx>0 i dy<0}
          a := a + 2 * pi;
        end
      end
    end
  end;

azg := a * rg; { obliczenie azymutu w gradach}
azs := a * rs; { obliczenie azymutu w stopniach}
writeln('dx = ', dx:10:3);
writeln('dy = ', dy:10:3);
writeln('azg [grad] =', azg:10:5); {Wydruk azymutu w grad z dokl: 10 poz. na
czesc calk. i 5 po kropce dzies}
writeln('azs [stopn]=' , azs:10:5);
writeln('Odleg z funkcji =', dlug(dx, dy):10:4); {Odleglosc obliczona z
funckcji dlug(dx,dy)}
writeln;

czekaj;

end; {Koniec procedury}

{----- Program glowny -----}

Begin {Program glowny}

koniec := 0;

clrscr;
GetDate(rok,mc,dz,dzt); {Sciagniecie ddaty z systemu - rok, miesciac, dzien,
dzien tygodnia}
writeln('Data: ',dz,' ',mc,' ',rok); {Wydruk daty}
writeln;
writeln(' Program DlugAz.pas');
writeln;
writeln;

while koniec <> 1 do {Dopoki zmienna koniec rozna od 1}

begin {Poczatek while }

repeat {Powtarzaj}
  writeln(' Obliczenie dlugosci lub azymutu ze wspolrzecznych');
  writeln;
  writeln(' 0 Koniec obliczen ');
  writeln(' 1 Dlugosci ze wspolrzecznych');
  writeln(' 2 Dlugosc z funkcji');
  writeln(' 3 Azymut ze wspolrzecznych');
  writeln;
  write('Wybierz wariant: 0..2 ==> ');
  readln(iw);
until (iw >= 0) and (iw <= 3); {Koniec powtarzaj az wprowadzone iw >=0 i <=
3}

CASE iw of {Przypadek iw}
  0: koniec := 1; {przypadek 0 - koniec =1 czyli wyjscie z programu}

```

```
1: Dlugosci; {Wywołanie procedury Dlugosci}  
2: Dlfun;    {Wywołanie procedury Dlfun}  
3: Azymut;   {Wywołanie procedury Azymut}
```

END

```
end;      {Koniec while - jesli koniec =1 to przejście do następnej linii  
ponizej, wyczyszczenie ekranu i wyjście}
```

```
clrscr;
```

```
End. {Koniec programu głównego}
```