

Triangle rectangle

Xcas

```
trirectangle():={
local xA,xB,xC,yA,yB,yC,a,b,c ;
saisir(xA,yA);
saisir(xB,yB);
saisir(xC,yC);
a:=(xB-xA)^2+(yB-yA)^2;
b:=(xC-xB)^2+(yC-yB)^2;
c:=(xC-xA)^2+(yC-yA)^2;
si (a==b+c) alors
  afficher ("rectangle en c");
fsi
si (b==a+c) alors
  afficher ("rectangle en B");
fsi;
si (c==b+a) alors
  afficher ("rectangle en A");
fsi ;
si (a!=b+c et b!=a+c et c!=a+b) alors
  afficher ("pas rectangle");
fsi;
}
```

ALGOBOX

```
▼ VARIABLES
  xA EST_DU_TYPE NOMBRE
  xB EST_DU_TYPE NOMBRE
  xC EST_DU_TYPE NOMBRE
  yA EST_DU_TYPE NOMBRE
  yB EST_DU_TYPE NOMBRE
  yC EST_DU_TYPE NOMBRE
  a EST_DU_TYPE NOMBRE
  b EST_DU_TYPE NOMBRE
  c EST_DU_TYPE NOMBRE
▼ DEBUT_ALGORITHME
  LIRE xA
  LIRE yA
  LIRE xB
  LIRE yB
  LIRE xC
  LIRE yC
  a PREND_LA_VALEUR pow(xB-xA,2)+pow(yB-yA,2)
  b PREND_LA_VALEUR pow(xC-xA,2)+pow(yC-yA,2)
  c PREND_LA_VALEUR pow(xC-xB,2)+pow(yC-yB,2)
  ▼ SI (c==a+b) ALORS
    DEBUT_SI
    AFFICHER "Le triangle est rectangle en A"
    FIN_SI
  ▼ SI (b==a+c) ALORS
    DEBUT_SI
    AFFICHER "Le triangle est rectangle en B"
    FIN_SI
  ▼ SI (a==b+c) ALORS
    DEBUT_SI
    AFFICHER "Le triangle est rectangle en C"
    FIN_SI
  ▼ SI (a!=b+c ET b!=a+c ET c!=a+b) ALORS
    DEBUT_SI
    AFFICHER "Le triangle n'est pas rectangle"
    FIN_SI
  FIN_ALGORITHME
```

CASIO

```
=====TRI RECT =====
"XA":?→X↵
"YA":?→Y↵
"XB":?→I↵
"YB":?→J↵
"XC":?→M↵
"YC":?→N↵
(X-I)²+(Y-J)²→C↵
(X-M)²+(Y-N)²→B↵
(I-M)²+(J-N)²→A↵
If C=A+B↵
Then "ABC RECT EN C"↵
IfEnd↵
If B=A+C↵
Then "ABC RECT EN B"↵
IfEnd↵
If A=B+C↵
Then "ABC RECT EN A"↵
IfEnd↵
If A≠B+C And B≠A+C And C≠A+B↵
Then "ABC NON RECT"↵
IfEnd
```

TEXAS

```
PROGRAM: TRI RECT
: Input "XA ",X
: Input "YA ",Y
: Input "XB ",I
: Input "YB ",J
: Input "XC ",M
: Input "YC ",N
: (X-I)²+(Y-J)²→C
: (X-M)²+(Y-N)²→B
: (I-M)²+(J-N)²→A
If C=A+B
:Then
:Disp "ABC RECT
EN C"
:End
:If B=A+C
:Then
:Disp "ABC RECT
EN B"
:End
:If A=B+C
:Then
:Disp "ABC RECT
EN A"
:End
:If A≠B+C et B≠A
+C et C≠A+B
:Then
:Disp "ABC NON R
ECT"
:End
```