

Fórmula SRK-T escrita en Delphi

Escrito por Manuel Diego Valdearenas Martin

Jueves, 04 de Mayo de 2017 12:43 - Actualizado Miércoles, 26 de Agosto de 2020 10:33

Fórmula SRK-T escrita en Delphi.

1. Calcular la refracción que obtendremos al implantar una LIO de una potencia determinada

```
////////////////////////////////////
//////// BIOMETRIA SRK-T 1990 A
// NECESITA k1 (potencia corneal), k2 (potencia corneal
meridiano complementario), alx (longitud axial ecografia A),
// A (Constante A LIO), V (distancia al vertice de
las gafas)
// DEVUELVE Devuelve la Refraccion Postoperatoria al
implantar una LIO de una potencia determinada en dioptrias (SINGLE)
// Journal Catataract Refractive Surgery Mayo 1990, pag.333-40
////////////////////////////////////
// Biometria
FUNCTION FN_REFRACCION_BIOMETRIA_SRK_T_LIO(VAR_k1,
VAR_k2, VAR_alx, VAR_Aconstante, VAR_LIO, VAR_V:
single): single; // SRK-T LIO --> Rx
var
K1: Single; // Potencia Corneal 1 en dioptrias
K2: Single; // Potencia Corneal 2 en dioptrias
ALx: Single; // Longitud Axial en mm
Aconstante: single; // Constante A LIO
LIO_Potencia: single; // Potencia Lente deseada
V: single; // Distancia al vertice
Na: single; // indice refraccion Humor acuoso
NC: single; // indice refraccion cornea
R1: single; // Radio de Curvatura corneal 1
R2: single; // Radio de Curvatura corneal 2
R: single; // Radio de Curvatura corneal Media
RE_THICK: single; // Grosor REtina
ACDSRK: single; // Constante ACD prara la formula SRK
ALxCOR: single; // Longitud Axial corregida para ojos
largos
Cw: single; // Anchura Corneal
H: single; // Altura Corneal
HH: single; // calculos
ELP: single; // Profundidad camara anterior
postoperatoria calculada
NCM1: single; // indice refraccion cornea menos 1 NC -1
Numerador: single; // numerador formula
Denominador: single; // denominador formula
Offset: single; // difference between IOL vs. natural lens
to cornea
LOPT: single; // Longitud axial optica
Es_Negativo: single;
Es_Negativo2: single;
```

Fórmula SRK-T escrita en Delphi

Escrito por Manuel Diego Valdearenas Martin

Jueves, 04 de Mayo de 2017 12:43 - Actualizado Miércoles, 26 de Agosto de 2020 10:33

```
Kav:           single;      // Potencia Corneal Media
begin
  K1            := VAR_k1;
  K2            := VAR_k2;
  ALx           := VAR_alx;
  Aconstante    := VAR_Aconstante;
  LIO_Potencia  := VAR_LIO;
  V             := VAR_V;

  Na            := 1.336;
  NC            := 1.333;

  R1            := 337.5 / K1;
  R2            := 337.5 / K2;
  R             := (R1 + R2) / 2;
  Kav           := 337.5 / R;

  RE_THICK      := 0.65696 - 0.02029 * ALx;
  LOPT          := ALx + RE_THICK;
  NCM1          := NC - 1;

  if (Aconstante > 100) then
    begin
      ACDSRK := Aconstante * 0.62467 - 68 - 0.74709;
    end
  else
    begin
      ACDSRK := Aconstante;
    end;
  Offset      := ACDSRK - 3.3357;

  if (ALx > 24.2) then
    begin
      ALxCOR := -3.446 + 1.716 * ALx - 0.0237 * sqr(ALx);
    end
  else
    begin
      ALxCOR := ALx;
    end;
  Cw          := -5.40948 + 0.58412 * ALxCOR + 0.098 * Kav;

  Es_Negativo := sqr(R) - ((sqr(Cw)) / 4);
  if Es_Negativo < 0 then
    begin
      Es_Negativo2 := 0;
    end
  else
    begin
```

```
Es_Negativo2 := Es_Negativo;
end;
H := R - sqrt(Es_Negativo2);
HH := H;
if HH > 5.5 then H := 5.5;

ELP := H + Offset;

Numerador := 1000 * Na * (Na * R - NCM1 * LOPT) - LIO_Potencia *
(LOPT - ELP) * (Na * R - NCM1 * ELP);
Denominador := (Na * (V * (Na * R - NCM1 * LOPT) + LOPT * R) - 0.001
* LIO_Potencia * (LOPT - ELP) * (V * (Na * R - NCM1 * ELP) + ELP *
R));
result := Numerador / Denominador;
end;
```

2. Calcular la potencia de la LIO que tenemos que implantar para conseguir una Refracción dinal RX, deseada

```
////////////////////////////////////
//////// 12 BIOMETRIA SRK-T 1990 RX
// NECESITA k1 (potencia corneal), k2 (potencia corneal
meridiano complementario), alx (longitud axial ecografia A),
// A (Constante A LIO), Rx (refraccion
postoperatoria deseada), V (distancia al vertice de las gafas)
// DEVUELVE Devuelve la LIO para una Refraccion
Postoperatoria deseada (RX) en dioptrias (SINGLE)
// Journal Catataract Refractive Surgery Mayo 1990, pag.333-40
////////////////////////////////////
// Biometria
FUNCTION FN_REFRACCION_BIOMETRIA_SRK_T_RX(VAR_k1, VAR_k2, VAR_alx,
VAR_Aconstante, VAR_Rx, VAR_V: single): single; //
SRK-T Rx ----> LIO
var
K1: Single; // Potencia Corneal 1 en dioptrias
K2: Single; // Potencia Corneal 2 en dioptrias
ALx: Single; // Longitud Axial en mm
Aconstante: single; // Constante A LIO
RX: single; // Refraccion Postoperatoria deseada
V: single; // Distancia al vertice
Na: single; // indice refraccion Humor acuoso
NC: single; // indice refraccion cornea
R1: single; // Radio de Curvadura corneal 1
R2: single; // Radio de Curvadura corneal 2
R: single; // Radio de Curvadura corneal Media
RE_THICK: single; // Grosor REtina
```

Fórmula SRK-T escrita en Delphi

Escrito por Manuel Diego Valdearenas Martin

Jueves, 04 de Mayo de 2017 12:43 - Actualizado Miércoles, 26 de Agosto de 2020 10:33

```
ACDSRK:      single;    // Constante ACD prara la formula SRK
ALxCOR:      single;    // Longitud Axial corregida para ojos
largos
  Cw:        single;    // Anchura Corneal
  H:         single;    // Altura Corneal
  HH:        single;    // calculos
  ELP:       single;    // Profundidad camara anterior
postoperatoria calculada
  NCM1:      single;    // indice refraccion cornea  menos 1 NC -1
  Numerador: single;    // numerador formula
  Denominador: single;  // denominador formula
  Offset:    single;    // difference between IOL vs. natural lens
to cornea
  LOPT:      single;    // Longitud axial optica
  Es_Negativo: single;
  Es_Negativo2: single;
  Kav:       single;    // Potencia Corneal Media
begin
  K1      := VAR_k1;
  K2      := VAR_k2;
  ALx     := VAR_alx;
  Aconstante := VAR_Aconstante;
  RX      := VAR_Rx;
  V       := VAR_V;

  Na      := 1.336;
  NC      := 1.333;

  R1      := 337.5 / K1;
  R2      := 337.5 / K2;
  R       := (R1 + R2) / 2;
  Kav     := 337.5 / R;

  RE_THICK := 0.65696 - 0.02029 * ALx;
  LOPT     := ALx + RE_THICK;
  NCM1     := NC - 1;

  if (Aconstante > 100) then
    begin
      ACDSRK := Aconstante * 0.62467 - 68 - 0.74709;
    end
  else
    begin
      ACDSRK := Aconstante;
    end;
  Offset := ACDSRK - 3.3357;

  if (ALx > 24.2) then
```

Fórmula SRK-T escrita en Delphi

Escrito por Manuel Diego Valdearenas Martin

Jueves, 04 de Mayo de 2017 12:43 - Actualizado Miércoles, 26 de Agosto de 2020 10:33

```
begin
    ALxCOR := -3.446 + 1.716 * ALx - 0.0237 * sqr(ALx);
end
else
begin
    ALxCOR := ALx;
end;
Cw      := -5.40948 + 0.58412 * ALxCOR + 0.098 * Kav;

Es_Negativo := sqr(R) - ((sqr(Cw)) / 4);
if Es_Negativo < 0 then
begin
    Es_Negativo2 := 0;
end
else
begin
    Es_Negativo2 := Es_Negativo;
end;
H := R - sqrt(Es_Negativo2);
HH := H;
if HH > 5.5 then H := 5.5;

ELP := H + Offset;

Numerador := 1000 * Na * (Na * R - NCM1 * LOPT - 0.001 * RX * (V
* (Na * R - NCM1 * LOPT) + LOPT * R));
Denominador := (LOPT - ELP) * (Na * R - NCM1 * ELP - 0.001 * RX *
(V * (Na * R - NCM1 * ELP) + ELP * R));
result := Numerador / Denominador;

end;
```