UNITED STATES
DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

QUADRATIC SPLINE SUBROUTINE PACKAGE

Water-Resources Investigations 82-41

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO.	2.	3. Recipient's Accession No.
4. Title and Subtitle Quadratic Spline Subroutine Package			5. Report Date Approved April 1982 6.
7. Author(s) L. A. Rasmussen			8. Performing Organization Rept. No. USGS/WRI 82-41
9. Performing Organization Name and Address U.S. Geological Survey, Water Resources Division Project Office - Glaciology 1201 Pacific Avenue, Suite 850 Tacoma WA 98402			10. Project/Task/Work Unit No. 11. Contract(C) or Grant(G) No. (C) (G)
12. Sponsoring Organization Name and Address			13. Type of Report & Period Covered 14.

15. Supplementary Notes

16. Abstract (Limit: 200 words)

A continuous piecewise quadratic function with continuous first derivative is devised for approximating a single-valued, but unknown, function represented by a set of discrete points. The quadratic is proposed as a treatment intermediate between using the angular (but reliable, easily constructed and manipulated) piecewise linear function and using the smoother (but occasionally erratic) cubic spline. Neither iteration nor the solution of a system of simultaneous equations is necessary to determining the coefficients. Several properties of the quadratic function are given.

A set of five short FORTRAN subroutines is provided for generating the coefficients (QSC), finding function value and derivatives (QSY), integrating (QSI), finding extrema (QSE), and computing arc length and the curvature-squared integral (QSK).

17. Document Analysis a. Descriptors

Computer programs, algorithms, Fortran

b. Identifiers/Open-Ended Terms

Quadratic spline, curve fitting, subroutines

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
No restriction on distribution	UNCLASSIFIED	16
	20. Security Class (This Page) UNCLASSIFIED	22. Price

(See ANSI-Z39.18) *See Instructions on Reverse* OPTIONAL FORM 272 (4-77) (Formerly NTIS-35) Department of Commerce

UNITED STATES
DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

QUADRATIC SPLINE SUBROUTINE PACKAGE

by L. A. Rasmussen

Water-Resources Investigations 82-41

Tacoma, Washington
1982

UNITED STATES DEPARTMENT OF THE INTERIOR

JAMES G. WATT, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information write to:

U.S. Geological Survey
Project Office - Glaciology
1201 Pacific Avenue - Suite 850
Tacoma, Washington 98402-4383

Table of Contents

QUADRATIC SPLINE SUBROUTINE PACKAGE

by L. A. Rasmussen

ABSTRACT

A continuous piecewise quadratic function with continuous first derivative is devised for approximating a single-valued, but unknown, function represented by a set of discrete points. The quadratic is proposed as a treatment intermediate between using the angular (but reliable, easily constructed and manipulated) piecewise linear function and using the smoother (but occasionally erratic) cubic spline. Neither iteration nor the solution of a system of simultaneous equations is necessary to determining the coefficients. Several properties of the quadratic function are given.

A set of five short FORTRAN subroutines is provided for generating the coefficients (QSC), finding function value and derivatives (QSY), integrating (QSI), finding extrema (QSE), and computing arc length and the curvature-squared integral (QSK).

INTRODUCTION

The continuous piecewise quadratic function with continuous derivative is a member of the class of functions known as polynomial splines of zero deficiency; that is, a piecewise polynomial of degree k with continuity of derivatives 0, 1, ..., k-1. The widely used cubic spline and the very widely used continuous piecewise linear function are both members of this class.

Although splines of even degree are little used directly, because of the asymmetry of the required boundary conditions, there is much justification for the existence of algorithms for constructing and manipulating quadratic splines. The derivative of a cubic spline is a quadratic spline, and the integral of a continuous piecewise linear function is a quadratic spline.

The quadratic spline generally has a jump discontinuity of the second derivative at the junction points, which in the function presented here are taken to be the data points. Although lacking continuity of the second derivative, it is free from the extraneous inflection points between data points, which sometimes occur with cubic splines (de Boor, 1978). The splines in tension, devised to avoid that defect, often concentrate the curvature near the data points (Ahlberg and others, 1967).

CONSTRUCTING THE SPLINE

Among the several criteria that could be used for generating a quadratic spline, the one chosen here leads to a compact, direct algorithm that does not require a boundary condition derivative. It maximizes the agreement of the quadratic spline with separately estimated values of the slopes at the given points.

Given y_i at each of $x_1<x_2<\ldots<x_n$ on a differentiable, single-valued, but unknown function $\phi(x)$, a piecewise quadratic $y=F(x)$ that is continuous in function value and first derivative is constructed so that between each of the n-1 pairs of successive points x_i, x_{i+1} the function is a separate quadratic

$$F(x) = \begin{cases} y_1 & (x = x_1) \\ f_i(x) & (x_i<x\leq x_{i+1}) \end{cases} \qquad (1)$$

where

$$f_i(x) \equiv a_i x^2 + b_i x + c_i \qquad (2)$$

which has slope

$$f_i'(x) \equiv df_i(x)/dx = 2a_i x + b_i \qquad (3)$$

The continuity conditions require for $2 \le i \le n-1$

$$\left.\begin{aligned} f_{i-1}(x_i) &= f_i(x_i) = y_i \\ f'_{i-1}(x_i) &= f'_i(x_i) \equiv s_i \end{aligned}\right\} \qquad (4)$$

where the slopes s_i are to be determined, and the boundary conditions require

$$\left.\begin{aligned} f_1(x_1) &= y_1 \\ f_{n-1}(x_n) &= y_n \end{aligned}\right\} \qquad (5)$$

Thus F(x) is a polynomial spline of degree 2 and deficiency zero.

In terms of the slope s_i, the coefficients of $f_i(x)$ are given, in turn, by

$$\left.\begin{aligned} a_i &= (R_i - s_i)/(x_{i+1} - x_i) \\ b_i &= s_i - 2a_i x_i \\ c_i &= y_i - (a_i x_i + b_i)x_i \end{aligned}\right\} \qquad (6)$$

where

$$R_i = (y_{i+1} - y_i)/(x_{i+1} - x_i) \qquad (7)$$

When equation (3) is applied at x_{i+1} to $f_i(x)$, in which a_i and b_i are taken from equation (6), there occurs the recurrence relation

$$s_{i+1} = 2R_i - s_i \qquad (8)$$

which can be used to give all subsequent s_i linearly in terms of s_1

$$s_i = (-1)^{i+1}\left[s_1 + 2\sum_{j=1}^{i-1}(-1)^j R_j\right] \qquad (9)$$

If equation (9) is written in the form

$$s_i = g_i s_1 + h_i \qquad (10)$$

in which $g_1=1$ and $h_1=0$, subsequent g_i and h_i obey the convenient recurrence relations

$$\left.\begin{aligned} g_{i+1} &= -g_i \\ h_{i+1} &= 2R_i - h_i \end{aligned}\right\} \qquad (11)$$

The construction of F(x) therefore has one degree of freedom, which through equations (6,9) may be expressed in terms of s_1. Chosen here as the property of F(x) to be optimized, since it leads to a closed form expression for s_1, is a measure of the agreement between the s_i and separately estimated values z_i of the slopes of $\phi(x)$ at the x_i. The z_i are obtained from n-2 auxiliary quadratics $\overline{f}_2(x)$, $\overline{f}_3(x)$, ..., $\overline{f}_{n-1}(x)$ where $\overline{f}_i(x)$ passes through (x_{i-1}, y_{i-1}), (x_i, y_i), and (x_{i+1}, y_{i+1}); it has coefficients

$$\overline{f}_i(x) \equiv \overline{a}_i x^2 + \overline{b}_i x + \overline{c}_i \tag{12}$$

which are given, in turn, by

$$\left.\begin{aligned} \overline{a}_i &= (R_i - R_{i-1})/(x_{i+1} - x_{i-1}) \\ \overline{b}_i &= R_i - (x_i + x_{i+1})\overline{a}_i \\ \overline{c}_i &= y_i - (\overline{a}_i x_i + \overline{b}_i) x_i \end{aligned}\right\} \tag{13}$$

The z_i are obtained by differentiating equation (12)

$$\overline{f}_i'(x) \equiv d\overline{f}_i(x)/dx = 2\overline{a}_i x + \overline{b}_i \tag{14}$$

and setting

$$z_i = \begin{cases} \overline{f}_2'(x_1) & (i=1) \\ \overline{f}_i'(x_i) & (2 \le i \le n-1) \\ \overline{f}_{n-1}'(x_n) & (i=n) \end{cases} \tag{15}$$

The quantity to be minimized is

$$I = \sum_{i=1}^{n} \left(\frac{s_i - z_i}{1 + z_i^2} \right)^2 \tag{16}$$

which, when substituting for s_i from equation (10) and requiring $\partial I/\partial s_1 = 0$, yields

$$s_1 = \sum_{i=1}^{n} \frac{g_i(z_i - h_i)}{(1 + z_i^2)^2} \bigg/ \sum_{i=1}^{n} \frac{1}{(1 + z_i^2)^2} \tag{17}$$

Finally, with s_1 determined, equations (6-8) are used to get the coefficients of the n-1 quadratics $f_i(x)$.

PROPERTIES

Equations (1-17) constitute a compact, direct algorithm for constructing $F(x)$, as neither numerical iteration nor the solution of a system of simultaneous linear equations is necessary. This is a consequence of the chosen norm, equation (16). Although closed form expressions for arc length

$$A \equiv \int_{x_1}^{x_n} \{1 + [dF(x)/dx]^2\}^{1/2}\, dx = \sum_{i=1}^{n-1} p_i$$

where

$$p_i = \begin{cases} \dfrac{1}{4a_i}\left[s\sqrt{1+s^2} + \ln(s+\sqrt{1+s^2})\right]\Big|_{s_i}^{s_{i+1}} & (a \neq 0) \\ (x_{i+1} - x_i)\sqrt{1 + b_i^2} & (a = 0) \end{cases} \qquad (18)$$

and for the integral of the square of curvature

$$C \equiv \int_{x_1}^{x_n} [d^2F(x)/dx^2]^2 \{1 + [dF(x)/dx]^2\}^{-3} dx = \sum_{i=1}^{n-1} q_i$$

where

$$q_i = \frac{a_i}{4}\{3\tan^{-1}s + s[3 + 2/(1 + s^2)]/(1 + s^2)\}\Big|_{s_i}^{s_{i+1}} \qquad (19)$$

may be written, minimizing either A or C requires solving a nonlinear equation in s_1.

The quantity in equation (16) roughly resembles the angle θ between $F(x)$ and the slope imputed through equations (13-15) to $\phi(x)$. From the identity

$$\tan\theta = \frac{s - z}{1 + sz}$$

may be obtained

$$\frac{s - z}{1 + z^2} = \frac{\tan\theta}{1 - z\tan\theta}$$

which, under the small angle approximation $\tan\theta \approx \theta$, shows that as the goodness of fit between the s_i and z_i increases, I tends to $\sum\theta^2$.

All the following properties can be derived from equations (1-17). Because those derivations cannot be expressed concisely here, and because the results can be easily verified experimentally, the properties are only stated.

If $\phi(x)$ is a polynomial of degree less than three, $F(x)$ is identical with $\phi(x)$.

If the x_i are equally spaced and if n is odd, the integral of $F(x)$ coincides with the result of applying Simpson's rule.

Because of the constant in the denominator of equation (16), $F(x)$ is not independent of scaling; that is, for the same set of x_i, where λ is a scalar, the $F(x)$ constructed from the ordinates $(\lambda y_1, \lambda y_2, \ldots, \lambda y_n)$ is not λ times the $F(x)$ constructed from the ordinates $(y_1, y_2, \ldots, y_n)$. Therefore, $F(x)$ is not additive; that is, for the same set of x_i, the $F(x)$ constructed from the ordinates $(y_1 + \hat{y}_1, y_2 + \hat{y}_2, \ldots, y_n + \hat{y}_n)$ is not the sum of the $F(x)$ constructed from the ordinates $(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)$ and the $F(x)$ constructed from the ordinates $(y_1, y_2, \ldots, y_n)$.

A convex $F(x)$ does not exist for certain convex (x_i, y_i). This, in fact, afflicts all quadratic splines, regardless of how S_1 may be chosen. If the R_i are monotonic, there may exist no S_i satisfying equation (8) that are monotonic. Consider the five points (0, 0), (10, 16), (20, 28), (30, 32), (40, 34), which from equation (7) give R_i = 1.6, 1.2, 0.4, 0.2; then, from equation (9), S_i = S_1, 3.2 - S_1, S_1 - 0.8, 1.6 - S_1, S_1 - 1.2. The requirement that the S_i be decreasing, because the R_i are decreasing, leads to the incompatible set of inequalities $S_1 > 1.6$, $S_1 < 2.0$, $S_1 > 1.2$, $S_1 < 1.4$ of which the first and last are contradictory.

$F(x)$ cannot be made periodic.

THE SUBROUTINES

The five subroutines presented here are only the nucleus of a quadratic spline subroutine package. A coefficient-forming subroutine simpler than QSC could be produced by requiring the user to supply the slope S_1; this would be consistent with the usual design of cubic spline generators, which require that two boundary condition derivatives be supplied. Subroutine QSC was designed for situations in which the user has no additional information for determining a slope. However, such a coefficient forming subroutine that requires S_1 to be supplied would be independent of scaling; it could also be used together with an optimization algorithm to form a quadratic spline with minimum curvature-squared integral or with minimum arc length. Other possibilities are subroutines for finding the intersections of a quadratic spline and a linear function, for integrating the product of two quadratic splines, for finding the point on a quadratic spline nearest a specified external point, etc. All of these suggested subroutines could be written to be compatible with the five included here, as long as they used the same data storage scheme.

In the coding of the subroutines, only a small subset of FORTRAN was used in the hope of permitting ready implementation on a wide variety of computer systems. Execution speed was sacrificed slightly to reduce storage requirements and to simplify use of the subroutines. They were written under the assumption that the stated restrictions will be observed; thus, they do not expend time and storage in checking the data supplied them.

The five subroutines are independent of each other, except that the coefficients generated by QSC may be used by any of the other, but none of the set directly uses any other. They rely on the system on which they are used to supply elementary function generators for real-valued square root, natural logarithm, and single-argument inverse tangent with result in radians; for these they assume FORTRAN FUNCTION subprograms named, respectively, SQRT, ALOG, and ATAN. On a large scale CDC computer their storage requirements in central memory words (decimal) is QSC 165, QSY 72, QSI 137, QSE 125, and QSK 124. The execution time of the coefficient forming subroutine QSC is proportional to n, which is true also for direct methods for forming cubic splines; but only about half as many operations are needed to obtain the quadratic coefficients as are needed to obtain the cubic coefficients.

The usage of the subroutines must follow

DIMENSION X(N), Y(N), Q(3,N), T(N)

N = $n>2$

X = $x_1<x_2<\ldots<x_n$

Y = $y_1, y_2, \ldots, y_n$

$$\left.\begin{aligned} Q(3,i) &= a_i \\ Q(2,i) &= b_i \\ Q(1,i) &= c_i \end{aligned}\right\} \quad 1\le i<n$$

Q(1,n), Q(2,n), Q(3,n) are used as temporary scratch storage by subroutine QSC.

<u>To form the coefficients of F(x)</u>

CALL QSC (N, X, Y, Q)

Input: N, X, Y

Output: Q

To find function value and derivatives of $F(x)$ at $x = U$

CALL QSY (N, X, Q, U, YU, YPU, YPPU)

Input: N, X, Q, U

Output: $YU = F(U)$

$YPU = dF(U)/dx$

$YPPU = d^2F(U)/dx^2$

Note: if $U<x_1$, then x_1 is used for U

if $U>x_n$, then x_n is used for U

To integrate $F(x)$ from $x = U$ to $x = V$

CALL QSI (N, X, Q, U, V, T, P)

Input: N, X, Q, U, V

Output: T, P

Note: if $U<x_1$, then x_1 is used for U

if $U>x_n$, then x_n is used for U

if $V<x_1$, then x_1 is used for V

if $V>x_n$, then x_n is used for V

Note: on the first call T(N) must be preset to zero, and subroutine QSI will form $T(i) = t_i$ for $1 \leq i \leq n-1$ and set $T(N) = 1$; on subsequent calls the T-array is used for speedy computation of the integral

Method:
$$P = \int_U^V F(x)dx = \int_{x_1}^V F(x)dx - \int_{x_1}^U F(x)dx$$

using
$$\int_{x_1}^x F(x)dx = t_i + w_i(x) \text{ for } x_i \leq x \leq x_{i+1}$$

where
$$t_i = \begin{cases} -w_1(x_1) & i=1 \\ \sum_{j=1}^{i-1} w_j(x_{j+1}) - \sum_{j=1}^{i} w_j(x_j) & 2 \leq i \leq n-1 \end{cases}$$

in which $$w_i(x) = \int_0^x f_i(x)dx = c_i x + b_i x^2/2 + a_i x^3/3$$

To find the extreme values of F(x)

CALL QSE (N, X, Y, Q, XMAX, YMAX, XMIN, YMIN)

Input: N, X, Y, Q

Output: XMAX = x_{max}

YMAX = $F(x_{max})$

XMIN = x_{min}

YMIN = $F(x_{min})$

where

$$F(x_{max}) \begin{cases} > F(x) & x_1 \le x < x_{max} \\ \ge F(x) & x_{max} < x \le x_n \end{cases}$$

$$F(x_{min}) \begin{cases} < F(x) & x_1 \le x < x_{min} \\ \le F(x) & x_{min} < x \le x_n \end{cases}$$

To compute the arc length and the curvature-squared integral of F(x)

CALL QSK (N, X, Q, A, C)

Input: N, X, Q

Output: A, C

Method: Equation (18) is used for the arc length A, and equation (19) is used for the curvature-squared integral C.

REFERENCES

Ahlberg, J. H., Nilson, E. N., and Walsh, J. L., 1967, The theory of splines and their applications: New York, Academic Press, 284 pages.

de Boor, Carl, 1978, A practical guide to splines: New York, Springer-Verlag, 392 pages.

Attachment A

```
      SUBROUTINE QSC (N,X,Y,Q)
C
C     FIND COEFFICIENTS  Q  OF N-1 PIECEWISE QUADRATIC WITH CONTINUOUS
C     FIRST DERIVATIVE, THROUGH  N  GIVEN POINTS (X,Y) WITH INCREASING X
C
      DIMENSION X(N),Y(N),Q(3,N)
      NM1=N-1
C                          FORM  R(I)  IN  Q(1,I), FORM  Z(I)  IN  Q(2,I)
C
      Q(1,1)=(Y(2)-Y(1))/(X(2)-X(1))
      DO 1400 I=2,NM1
      DX=X(I+1)-X(I)
      Q(1,I)=(Y(I+1)-Y(I))/DX
      ABAR=(Q(1,I)-Q(1,I-1))/(X(I+1)-X(I-1))
      BBAR=Q(1,I)-ABAR*(X(I)+X(I+1))
      Q(2,I)=2.*ABAR*X(I)+BBAR
      IF (I-2) 1100,1100,1200
 1100 Q(2,1)=2.*ABAR*X(1)+BBAR
 1200 IF (I-NM1) 1400,1300,1300
 1300 Q(2,N)=2.*ABAR*X(N)+BBAR
 1400 Q(1,I)=(Y(I+1)-Y(I))/DX
C
C                          DETERMINE OPTIMUM VALUE OF  S(1)
      U=0.
      V=0.
      G=1.
      H=0.
      DO 2200 I=1,N
      W=1./(1.+Q(2,I)**2)**2
      U=U+G*(Q(2,I)-H)*W
      V=V+W
      IF (I-N) 2100,2300,2300
 2100 G=-G
 2200 H=2.*Q(1,I)-H
 2300 S=U/V
C                          FORM  N-1  SETS  OF  QUADRATIC  COEFFICIENTS
C                          C(I) IN Q(1,I), B(I) IN Q(2,I), A(I) IN Q(3,I)
      DO 3100 I=1,NM1
      Q(3,I)=(Q(1,I)-S)/(X(I+1)-X(I))
      Q(2,I)=S-2.*X(I)*Q(3,I)
      S=2.*Q(1,I)-S
 3100 Q(1,I)=Y(I)-X(I)*(Q(2,I)+X(I)*Q(3,I))
C
      RETURN
      END
```

Attachment B

```
      SUBROUTINE QSE (N,X,Y,Q,XMAX,YMAX,XMIN,YMIN)
C
C     GIVEN STORED IN  Q  THE N-1 PIECEWISE QUADRATIC Y=F(X) THAT PASSES
C     THROUGH  N  POINTS (X,Y) WITH INCREASING  X, FIND (XMAX,YMAX) THE
C     POINT WITH MAXIMUM  Y, AND (XMIN,YMIN) THE POINT WITH MINIMUM  Y.
C     OF TWO POINTS WITH THE SAME  Y, THE ONE WITH SMALLER  X  IS CHOSEN
C
      DIMENSION X(N),Y(N),Q(3,N)
      NM1=N-1
C                  ASSUME  Y(1)  IS BOTH GLOBAL MAXIMUM AND GLOBAL MINIMUM
      YMAX=Y(1)
      XMAX=X(1)
      YMIN=Y(1)
      XMIN=X(1)
C                  CHECK EACH OF  N-1  INTERVALS FOR EXTREME VALUES
C
      DO 1000 I=1,NM1
C                        CHECK IF LAST POINT IN INTERVAL IS EXTREME SO FAR
C
      IF (Y(I+1)-YMAX) 200,200,100
  100 YMAX=Y(I+1)
      XMAX=X(I+1)
      GO TO 400
  200 IF (Y(I+1)-YMIN) 300,400,400
  300 YMIN=Y(I+1)
      XMIN=X(I+1)
C                        DETERMINE IF EXTREMUM OF I-TH QUADRATIC IS IN THE
C                        INTERVAL AND, IF SO, IF IS EXTREME VALUE SO FAR
C
  400 IF (Q(3,I)) 500,1000,500
  500 XE=-0.5*Q(2,I)/Q(3,I)
      IF ((XE-X(I))*(X(I+1)-XE)) 1000,1000,600
  600 YE=Q(1,I)+XE*(Q(2,I)+XE*Q(3,I))
      IF (YE-YMAX) 800,1000,700
  700 YMAX=YE
      XMAX=XE
      GO TO 1000
  800 IF (YE-YMIN) 900,1000,1000
  900 YMIN=YE
      XMIN=XE
C
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE QSI (N,X,Q,U,V,T,P)
C
C     GIVEN STORED IN  Q  THE N-1 PIECEWISE QUADRATIC Y=F(X) THAT PASSES
C     THROUGH  N  POINTS (X,Y) WITH INCREASING  X, AND GIVEN U  AND  V,
C     FIND THE INTEGRAL  P  OF F(X) FROM X=U TO X=V.  IF U OR V IS NOT
C     IN CLOSED INTERVAL X(1),X(N) IT IS CORRECTED TO NEARER ENDPOINT.
C
      DIMENSION X(N),Q(3,N),T(N)
      NM1=N-1
      UU=AMAX1(X(1),AMIN1(X(N),U))
      VV=AMAX1(X(1),AMIN1(X(N),V))
      IF (T(N)) 2000,1000,2000
C
C                          FORM T-ARRAY (FIRST TIME CALLED WITH THIS Q)
 1000 T(1)=0.
      DO 1100 I=1,NM1
      T(I)=T(I)-X(I)*(Q(1,I)+X(I)*(0.5*Q(2,I)+X(I)*Q(3,I)/3.))
 1100 T(I+1)=T(I)+X(I+1)*(Q(1,I)+X(I+1)*(0.5*Q(2,I)+X(I+1)*Q(3,I)/3.))
      T(N)=1.
C                          INTEGRAL OF  F(X)  FROM  X(1)  TO  U
 2000 DO 2200 I=1,NM1
      IF (UU-X(I+1)) 2100,2100,2200
 2100 PU=T(I)+UU*(Q(1,I)+UU*(0.5*Q(2,I)+UU*Q(3,I)/3.))
      GO TO 3000
 2200 CONTINUE
C                          INTEGRAL OF  F(X)  FROM  X(1)  TO  V
 3000 DO 3200 I=1,NM1
      IF (VV-X(I+1)) 3100,3100,3200
 3100 PV=T(I)+VV*(Q(1,I)+VV*(0.5*Q(2,I)+VV*Q(3,I)/3.))
      GO TO 4000
 3200 CONTINUE
C                          INTEGRAL OF  F(X)  FROM  U  TO  V
 4000 P=PV-PU
C
      RETURN
      END
```

```
      SUBROUTINE QSK (N,X,Q,A,C)
C
C     GIVEN STORED IN  Q  THE N-1 PIECEWISE QUADRATIC Y=F(X) THAT PASSES
C     THROUGH  N  POINTS (X,Y) WITH INCREASING X, FIND FROM X(1) TO X(N)
C     BOTH THE ARC LENGTH AND THE INTEGRAL OF THE CURVATURE SQUARED.
C
      DIMENSION X(N),Q(3,N),DA(2),DC(2)
      NM1=N-1
      A=0.
      C=0.
C
      DO 4000 I=1,NM1
      IF (Q(3,I)) 2000,1000,2000
 1000 A=A+(X(I+1)-X(I))*SQRT(1.+Q(2,I)**2)
      GO TO 4000
C
 2000 II=I
      DO 3000 J=1,2
      Y=2.*Q(3,I)*X(II)+Q(2,I)
      U=1.+Y*Y
      V=SQRT(U)
      DC(J)=3.*ATAN(Y)+(3.+2./U)*Y/U
      DA(J)=ALOG(Y+V)+Y*V
 3000 II=II+1
C
      C=C+0.25*(DC(2)-DC(1))*Q(3,I)
      A=A+0.25*(DA(2)-DA(1))/Q(3,I)
 4000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE QSY (N,X,Q,U,YU,YPU,YPPU)
C
C     GIVEN X=U AND N-1 PIECEWISE QUADRATIC Y=F(X) STORED IN  Q  AND
C     PASSING THROUGH  N  POINTS (X,Y) WITH INCREASING  X, FIND AT  U
C     ORDINATE YU=F(U), FIRST DERIVATIVE YPU, AND SECOND DERIVATIVE YPPU
C     IF  U  IS NOT IN CLOSED INTERVAL X(1),X(N) NEARER ENDPOINT IS USED
C
      DIMENSION X(N),Q(3,N)
      UU=AMAX1(X(1),AMIN1(X(N),U))
C
      DO 2000 I=2,N
      IF (UU-X(I)) 1000,1000,2000
C
 1000 YU=Q(1,I-1)+UU*(Q(2,I-1)+UU*Q(3,I-1))
      YPPU=2.*Q(3,I-1)
      YPU=UU*YPPU+Q(2,I-1)
      GO TO 9000
C
 2000 CONTINUE
C
 9000 RETURN
      END
```