Acta Universitatis Palackianae Olomucensis. Facultas Rerum Naturalium. Mathematica

Josef Drdla; Václav Snášel
Data type polygon

DATA TYPE POLYGON

JOSEF DRDLA, VÁCLAV SNÁŠEL

(Received April 30, 1989)

In this article we deal with the definition of data struc-
ture polygon. This data structure is used in many applications,
such as interactive work at modeling curves, surfaces, in prob-
lems of discrete mathematics, in the theory of lattices, in the
theory of graphs and ordered sets.

In the first part the axiomatic definition of this struc-
ture will be given and in the second part we will outline the
way of implementation in computer graphics. During the last ten
years the development of programming aids showed qualitative
change. The centre of attention moved from the development of
isolated programming implements (compilers, linkers,...) to the
development of integrated programming systems and so called
programming environment. This change of approach needed new
methods in specifications of programs. One of the basic method
is the method of algebraic specifications. This method will be
used at design of type polygon, which should be applicable in
various integrated programming aids.

The description of type polygon will be made by algebraic specification.

Data-type Polygon
    Operations

                        NIL : Polygon
              add-vertice : V x polygon ⟶ polygon
              del-vertice : V x polygon ⟶ polygon
            empty-polygon : polygon ⟶ boolean
           active-vertice : V x polygon ⟶ polygon
       not-active-vertice : V x polygon ⟶ polygon
                make-edge : V x polygon ⟶ polygon
                 del-edge : V x polygon ⟶ polygon

    Axioms

            empty-polygon(NIL) = true
            empty-polygon(add-vertice(V,P)) = false
            add-vertice(V,(del-vertice(V,P))) = P
            active-vertice(V,(not-active-vertice(V,P))) = P
            make-edge(V,(del-edge(V,P))) = P

Data-type Polygon
Basic characteristic of individual operations :
NIL :
Standard notion of empty polygon.
add-vertice :
Adds a new point to polygon.
del-vertice :
Deletes the point in polygon and edges containing this point.
empty-polygon :
Function, which has a walue true if polygon is empty.
active-vertice :
Sets flag of activity of the given point.
not-active-vertice :
Deletes flag of activity of the given point.
make-edge :
Inserts to polygon edges connecting the given point with active points.

del-edge :
Deletes edges connecting the given point with active points.

We verified the application of **data type polygon on in**-teractive system of creation control polygon for Bezier´s method of modeling plane curves. Except procedures of data representation of polygon which were described in preceding part, the system contains graphics procedures working with the given type on monitor. These procedures must fulfil the demand of interactive work with polygon. It means not only the ability to display the given polygon but also to have possibility to flexiblely change the position of arbitrary point and so the shape of arbitrary part of polygon. The whole system of displaing and changing of shape polygon is directed by a mouse and works in three modes:
1) The mode of creating polygon.
2) The mode of changing shape polygon.
3) The mode of setting of supporting points polygon.

Ad 1) By moving the mouse cursor on monitor it is possible to create the basic shape of polygon when pressing the mouse button the edge of actual setting cursor to last active vertex of polygon is displayed.

Ad 2) We set the mouse cursor to the vertex of polygon, the position of which we want to change and we "catch" it by pressing the mouse button. Moving mouse we change the setting of polygon edges which go through the given vertex. Releasing the button we set out the new setting of vertex and corresponding edges.

Ad 3) From the definition of Bezier´s method it is evident, that n vertices of control polygon determine the curve of grade n-1 and by change of setting of the only vertex the whole shape of creating curve is changed. Just for the ability of local changes of curve we established the so-called supporting points which can be set out by mouse to selected vertices of the given polygon. These point will divide the whole polygon into individual parts, whe each isapproximated by Be-

zier´s method separately and created curves go through the
given supporting points.

Graphic procedures can be divided into three groups:

1) Attribute setting procedures:
procedure set-vertice-type (Vtype, Vwidth, Vcolor : word)
Sets type, width and color of vertices of control polygon.

procedure set-edge-type (Etype, Ewidth, Ecolor : word)
Sets type, width and color of edges of control polygon.

procedure set-mouse-type (Mtype : word)
Sets type the mouse cursor.

For creating polygon we choose cross type cursor, for the
change of the shape polygon and setting supporting points we
choose a ring.

2) Drawing procedures

procedure draw-vertex (X,Y : integer)
Displais vertex of the given type, when X,Y are coordinates
vertex on monitor.

procedure draw-edge (X,Y : integer)
Displais edge of the given type, when X,Y are coordinates of
terminal point of edge, which comes from active vertex of
polygon.

3) Procedures working with placement of mouse cursor

procedure set-mouse-XY (X,Y : integer)

Sets mouse cursor to the given place on monitor,
(X,Y ... coordinate mouse cursor).

procedure get-mouse-button XY (var X,Y,B : integer)
Determines the setting of mouse cursor on monitor and
pressing or releasing of any mouse button,
(X,Y ... coordinate mouse cursor, B ... status mouse button).

SOUHRN

DATOVÝ TYP POLYGON

JOSEF DRDLA, VÁCLAV SNÁŠEL

Naším hlavním cílem byl popis typu polygon. V článku je
tento typ popsán pomocí algebraické specifikace a dále je zde
popsána geometrická realizace.

РЕЗЮМЕ

ДАТОВЫЙ ТИП ПОЛИГОН

Й. ДРДЛА, В. СНАШЕЛ

Нашей главной целью было дать описание типа полигон.
В статье рассматривается алгебраическая спецификация и
геометрическая реализация типа полигон.

REFERENCES

[1]   F u t i, K. - S u z u k i, N.: Programming Language and VLSI scheme.
      Mir 1988
[2]   W o l f g a n g, G.: Interactive Computer Graphics. Prentice-Hall
      1978

Author´s address:
RNDr.Josef Drdla
RNDr.Václav Snášel
Katedra výpočetní techniky PčF UP
771 46 Olomouc
Czechoslovakia