

ELLIPSOID

10xxxxxx.1

LEGAL RESTRICTIONS:

This Reusable Software Component (RSC) contains data with Unlimited Government Rights.

DESCRIPTION:

ELLIPSOID is a C language code component that provides access to ellipsoid parameters for a large collection of ellipsoids.

CERTIFICATION LEVEL:

This RSC has been certified at level 4. A level 4 component satisfies the criteria for reliability, testing, and documentation for the Army Reuse Center (ARC). The component comes with test materials and a Reuse Manual that aids in integrating the component into a software system.

LEVEL OF TESTING/ACCEPTANCE:

Unit and integration testing have been performed for the functions contained in this component.

PURPOSE/INTENDED USE:

The purpose of ELLIPSOID is to provide access to ellipsoid parameters for a collection of common ellipsoids. A particular ellipsoid can be accessed by using its standard 2-letter code to find its index in the ellipsoid table. The index can then be used to retrieve the ellipsoid name and parameters.

By sequentially retrieving all of the ellipsoid codes and/or names, a menu of the available ellipsoids can be constructed. The index values resulting from selections from this menu can then be used to access the parameters of the selected ellipsoid.

This component depends on a data file named "ellips.dat", which contains the ellipsoid parameter values. A copy of this file must be located in the directory specified by the value of

the environment variable "ELLIPSOID_DATA", if defined, or else in the current directory, whenever a program containing this component is executed.

Additional ellipsoids can be added to this file, either manually or using the Create_Ellipsoid function. However, if a large number of ellipsoids are added, the ellipsoid table array size in this component will have to be increased.

HARDWARE/ENVIRONMENT CONSTRAINTS:

There are no hardware or environment constraints. There are no limitations.

FUNCTIONS:

Initialize_Ellipsoids – This function creates the ellipsoid table from an external file.

Create_Ellipsoid – This function creates a new ellipsoid with the specified code, name, and axes, adds it to the ellipsoid table, and updates the ellipsoid data file.

Ellipsoid_Count – This function returns the number of ellipsoids in the ellipsoid table.

Ellipsoid_Index – This function returns the index of the ellipsoid with the specified 2-letter code.

Ellipsoid_Name – This function returns the name of the ellipsoid referenced by index.

Ellipsoid_Code – This function returns the 2-letter code of the ellipsoid referenced by index.

Ellipsoid_Axes – This function returns the semi-major and semi-minor axis values for the ellipsoid referenced by index.

Ellipsoid_Eccentricity2 – This function returns the square of the eccentricity value for the ellipsoid referenced by index.

Ellipsoid_Flattening – This function returns the flattening value for the ellipsoid referenced by index.

WGS84_Axes – This function returns the semi-major and semi-minor axis values for the WGS84 ellipsoid.

WGS84_Eccentricity2 – This function returns the square of the eccentricity value for the WGS84 ellipsoid.

WGS84_Flattening – This function returns the flattening value for the WGS84 ellipsoid.

WGS72_Axes – This function returns the semi-major and semi-minor axis values for the WGS72 ellipsoid.

WGS72_Eccentricity2 – This function returns the square of the eccentricity value for the WGS72 ellipsoid.

WGS72_Flattening – This function returns the flattening value for the WGS72 ellipsoid.

EXAMPLE APPLICATIONS:

The following example illustrates how ELLIPSOID can be used to retrieve information about an ellipsoid.

Function Call:

```
status = Initialize_Ellipsoids()
```

Inputs:

none.

Outputs:

none.

Function Call:

```
status = Ellipsoid_Index(code, index)
```

Inputs:

code	"ED"
------	------

Outputs:

index:	11
--------	----

Function Call:

```
status = Ellipsoid_Name(index, name)
```

Inputs:

index:	11
--------	----

Outputs:

name: "Everest 1969 (West Malasia)"

Function Call:

status = Ellipsoid_Axes(index, a,b)

Inputs:

index: 11

Outputs:

a: 6377295.664

b: 6356094.668

Function Call:

status = Ellipsoid_Eccentricity2(index, e2)

Inputs:

index: 11

Outputs:

e2: 0.00663784660369520500

Function Call:

status = Ellipsoid_Flattening(index, f)

Inputs:

index: 11

Outputs:

f: 0.00332444929666288500