**XYZ\_IO   
XYZ File Input/Output routines.**

**XYZ\_IO** is a FORTRAN90 library which reads and writes files in the XYZ, XYZL and XYZF formats.

XYZ files are a simple way of storing information about sets of points in 3D.

An XYZ file has a simple structure. There are three kinds of records:

* COMMENT LINES begin with the character '#' and are ignored;
* BLANK LINES are ignored;
* COORDINATE LINES each contain one triple of XYZ coordinates;

Here is an example of an XYZ file containing 13 points:

# cube.xyz

#

0.000000 0.000000 0.000000

1.000000 0.000000 0.000000

1.000000 1.000000 0.000000

0.000000 1.000000 0.000000

0.000000 0.000000 1.000000

1.000000 0.000000 1.000000

1.000000 1.000000 1.000000

0.000000 1.000000 1.000000

Of course, in many cases, we would like to describe LINES between the points. We can add this information by including a second file, called an XYZL file, which consists of a string of point indices. A line is drawn from one point to the next, and so on, through all the points index on the record. Thus, an XYZL file associated with the above XYZ file might be:

# cube.xyzl

#

1 2 3 4 1

5 6 7 8 5

1 5

2 6

3 7

4 8

We might also like to describe FACES defined by the points. We can add this information by including a file, called an XYZF file, which consists of a string of point indices. A face is defined by listing in counterclockwise order the vertices that lie on it. The first vertex is not repeated as the final vertex. Thus, an XYZF file associated with the above XYZ file might be:

# cube.xyzf

#

1 4 3 2

2 3 7 6

5 6 7 8

5 8 4 1

1 2 6 5

3 4 8 7

**Licensing:**

The computer code and data files described and made available on this web page are distributed under [the GNU LGPL license.](https://people.sc.fsu.edu/~jburkardt/txt/gnu_lgpl.txt)

**Languages:**

**XYZ\_IO** is available in [a C++ version](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyz_io/xyz_io.html) and [a FORTRAN90 version](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io.html) and [a MATLAB version](https://people.sc.fsu.edu/~jburkardt/m_src/xyz_io/xyz_io.html).

**Related Data and Programs:**

[IVREAD](https://people.sc.fsu.edu/~jburkardt/f_src/ivread/ivread.html), a FORTRAN90 program which can convert graphics information between various 3D formats, including the XYZ format.

[PDB\_TO\_XYZ](https://people.sc.fsu.edu/~jburkardt/f_src/pdb_to_xyz/pdb_to_xyz.html), a FORTRAN90 program which reads the ATOM records from a PDB file, and writes the atomic coordinates to an XYZ file.

[SPHERE\_VORONOI](https://people.sc.fsu.edu/~jburkardt/f_src/sphere_voronoi/sphere_voronoi.html), a FORTRAN90 library which reads an XYZ file of 3D points on the unit sphere, and computes and plots Delaunay triangulations and Voronoi diagrams.

[XY](https://people.sc.fsu.edu/~jburkardt/data/xy/xy.html), a data directory which contains examples of XY files, a simple 2D graphics point and line format;

[XYZ](https://people.sc.fsu.edu/~jburkardt/data/xyz/xyz.html), a data directory which contains examples of XYZ files, a simple 3D graphics point and line format;

[XYZ\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyz_display/xyz_display.html), a MATLAB program which reads XYZ information defining points in 3D, and displays an image in the MATLAB graphics window.

[XYZ\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyz_display_opengl/xyz_display_opengl.html), a C++ program which reads an XYZ file of 3D point coordinates, and displays an image of those points using OpenGL.

[XYZ\_TO\_PDB](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_to_pdb/xyz_to_pdb.html), a FORTRAN90 program which reads a set of XYZ spatial coordinates, and rewrites them as ATOM records in a PDB file.

[XYZF](https://people.sc.fsu.edu/~jburkardt/data/xyzf/xyzf.html), a data directory which contains examples of XYZF files, a simple 3D graphics point and face format;

[XYZF\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyzf_display/xyzf_display.html), a MATLAB program which reads XYZF information defining points and faces in 3D, and displays an image in a MATLAB graphics window.

[XYZF\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyzf_display_opengl/xyzf_display_opengl.html), a C++ program which reads XYZL information defining points and faces in 3D, and displays an image using OpenGL.

[XYZL](https://people.sc.fsu.edu/~jburkardt/data/xyzl/xyzl.html), a data directory which contains examples of XYZL files, a simple 3D graphics point and line format;

[XYZL\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyzl_display/xyzl_display.html), a MATLAB program which reads XYZL information defining points and lines in 3D, and displays an image in a MATLAB graphics window.

[XYZL\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyzl_display_opengl/xyzl_display_opengl.html), a C++ program which reads XYZL information defining points and lines in 3D, and displays an image using OpenGL.

**Source Code:**

* [xyz\_io.f90](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io.f90), the source code.

**Examples and Tests:**

* [xyz\_io\_prb.f90](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io_prb.f90), a sample calling program.
* [xyz\_io\_prb\_output.txt](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io_prb_output.txt), the output file.
* [helix.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/helix.xyz), an XYZ file created by the example program.
* [xy\_io\_prb\_02.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xy_io_prb_02.xyz), an XYZ file created by the example program.
* [cube.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyz), an XYZ file of points created by the example program.
* [cube.xyzl](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyzl), an XYZL file of lines created by the example program.
* [cube.xyzf](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyzf), an XYZF file of lines created by the example program.

**List of Routines:**

* **CH\_CAP** capitalizes a single character.
* **CH\_EQI** is a case insensitive comparison of two characters for equality.
* **CH\_TO\_DIGIT** returns the integer value of a base 10 digit.
* **GET\_UNIT** returns a free FORTRAN unit number.
* **S\_TO\_I4** reads an integer value from a string.
* **S\_TO\_I4VEC** reads an integer vector from a string.
* **S\_TO\_R8** reads an R8 value from a string.
* **S\_TO\_R8VEC** reads an R8VEC from a string.
* **S\_WORD\_COUNT** counts the number of "words" in a string.
* **TIMESTAMP** prints the current YMDHMS date as a time stamp.
* **TIMESTRING** writes the current YMDHMS date into a string.
* **WORD\_NEXT\_READ** "reads" words from a string, one at a time.
* **XYZ\_DATA\_PRINT** prints the data of an XYZ file.
* **XYZ\_DATA\_READ** reads data from an XYZ file.
* **XYZ\_DATA\_WRITE** writes the data of an XYZ file.
* **XYZ\_EXAMPLE** computes points for an example XYZ dataset.
* **XYZ\_EXAMPLE\_SIZE** sizes an example XYZ dataset.
* **XYZ\_HEADER\_PRINT** prints the header of an XYZ file.
* **XYZ\_HEADER\_READ** determines the number of pairs of data in an XYZ file.
* **XYZ\_HEADER\_WRITE** writes the header of an XYZ file.
* **XYZ\_READ** reads graphics information from an XYZ file.
* **XYZ\_WRITE** writes an XYZ file.
* **XYZ\_WRITE\_TEST** tests the XYZ write routines.
* **XYZF\_DATA\_PRINT** prints the data of an XYZF file.
* **XYZF\_DATA\_READ** reads the data in an XYZF file.
* **XYZF\_DATA\_WRITE** writes the data of an XYZF file.
* **XYZF\_EXAMPLE** sets data suitable for a pair of XYZ and XYZF files.
* **XYZF\_EXAMPLE\_SIZE** sizes the data to be created by XYZF\_EXAMPLE.
* **XYZF\_HEADER\_PRINT** prints the header of an XYZF file.
* **XYZF\_HEADER\_READ** determines the number of face items in an XYZF file.
* **XYZF\_HEADER\_WRITE** writes the header of an XYZF file.
* **XYZF\_WRITE** writes an XYZF file.
* **XYZL\_DATA\_PRINT** prints the data of an XYZL file.
* **XYZL\_DATA\_READ** reads the data in an XYZL file.
* **XYZL\_DATA\_WRITE** writes the data of an XYZL file.
* **XYZL\_EXAMPLE** sets data suitable for a pair of XYZ and XYZL files.
* **XYZL\_EXAMPLE\_SIZE** sizes the data to be created by XYZL\_EXAMPLE.
* **XYZL\_HEADER\_PRINT** prints the header of an XYZL file.
* **XYZL\_HEADER\_READ** determines the number of line items in an XYZL file.
* **XYZL\_HEADER\_WRITE** writes the header of an XYZL file.
* **XYZL\_WRITE** writes an XYZL file.

You can go up one level to [the FORTRAN90 source codes](https://people.sc.fsu.edu/~jburkardt/f_src/f_src.html).

*Last revised on 08 January 2009.*

**XYZ\_IO   
XYZ文件输入/输出例程。**

**XYZ\_IO** 是一个FORTRAN90库，它以XYZ，XYZL和XYZF格式读写文件。

XYZ文件是一种存储3D点集信息的简单方法。

XYZ文件具有简单的结构。有三种记录：

* 评论线以字符“＃”开头并被忽略;
* BLANK LINES被忽略;
* 坐标线每个包含一个XYZ坐标的三倍;

以下是包含13个点的XYZ文件示例：

＃cube.xyz

＃

0.000000 0.000000 0.000000

1.000000 0.000000 0.000000

1.000000 1.000000 0.000000

0.000000 1.000000 0.000000

0.000000 0.000000 1.000000

1.000000 0.000000 1.000000

1.000000 1.000000 1.000000

0.000000 1.000000 1.000000

当然，在很多情况下，我们想描述各点之间的LINES。我们可以通过包含第二个文件来添加此信息，该文件称为XYZL文件，该文件由一串点索引组成。从一个点到下一个点绘制一条线，依此类推，通过记录上的所有点索引。因此，与上述XYZ文件关联的XYZL文件可能是：

＃cube.xyzl

＃

1 2 3 4 1

5 6 7 8 5

1 5

2 6

3 7

4 8

我们可能还想描述由这些点定义的FACES。我们可以通过包含一个名为XYZF文件的文件来添加此信息，该文件由一串点索引组成。通过以逆时针方式列出位于其上的顶点来定义面。第一个顶点不会重复作为最终顶点。因此，与上述XYZ文件关联的XYZF文件可能是：

＃cube.xyzf

＃

1 4 3 2

2 3 7 6

5 6 7 8

5 8 4 1

1 2 6 5

3 4 8 7

**许可：**

在此网页上描述和提供的计算机代码和数据文件是在[GNU LGPL许可](https://people.sc.fsu.edu/~jburkardt/txt/gnu_lgpl.txt)下分发 [的。](https://people.sc.fsu.edu/~jburkardt/txt/gnu_lgpl.txt)

**语言：**

**XYZ\_IO**是提供 [一个C ++版本](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyz_io/xyz_io.html)和 [一个FORTRAN90版本](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io.html)和 [一个MATLAB版本](https://people.sc.fsu.edu/~jburkardt/m_src/xyz_io/xyz_io.html)。

**相关数据和程序：**

[IVREAD](https://people.sc.fsu.edu/~jburkardt/f_src/ivread/ivread.html)，FORTRAN90程序，可以在各种3D格式之间转换图形信息，包括XYZ格式。

[PDB\_TO\_XYZ](https://people.sc.fsu.edu/~jburkardt/f_src/pdb_to_xyz/pdb_to_xyz.html)，FORTRAN90程序，从PDB文件读取ATOM记录，并将原子坐标写入XYZ文件。

[SPHERE\_VORONOI](https://people.sc.fsu.edu/~jburkardt/f_src/sphere_voronoi/sphere_voronoi.html)，FORTRAN90库，读取单位球面上3D点的XYZ文件，并计算和绘制Delaunay三角剖分和Voronoi图。

[XY](https://people.sc.fsu.edu/~jburkardt/data/xy/xy.html)，一个数据目录，包含XY文件的示例，简单的2D图形点和线格式;

[XYZ](https://people.sc.fsu.edu/~jburkardt/data/xyz/xyz.html)，一个数据目录，包含XYZ文件的示例，简单的3D图形点和线格式;

[XYZ\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyz_display/xyz_display.html)，一个MATLAB程序，读取定义3D点的XYZ信息，并在MATLAB图形窗口中显示图像。

[XYZ\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyz_display_opengl/xyz_display_opengl.html)，一个C ++程序，读取3D点坐标的XYZ文件，并使用OpenGL显示这些点的图像。

[XYZ\_TO\_PDB](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_to_pdb/xyz_to_pdb.html)，一个FORTRAN90程序，它读取一组XYZ空间坐标，并将它们重写为PDB文件中的ATOM记录。

[XYZF](https://people.sc.fsu.edu/~jburkardt/data/xyzf/xyzf.html)，一个数据目录，包含XYZF文件的示例，简单的3D图形点和面部格式;

[XYZF\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyzf_display/xyzf_display.html)，一个MATLAB程序，读取以3D形式定义点和面的XYZF信息，并在MATLAB图形窗口中显示图像。

[XYZF\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyzf_display_opengl/xyzf_display_opengl.html)，一个C ++程序，读取定义3D中的点和面的XYZL信息，并使用OpenGL显示图像。

[XYZL](https://people.sc.fsu.edu/~jburkardt/data/xyzl/xyzl.html)，一个包含XYZL文件示例的数据目录，一个简单的3D图形点和线格式;

[XYZL\_DISPLAY](https://people.sc.fsu.edu/~jburkardt/m_src/xyzl_display/xyzl_display.html)，一个MATLAB程序，读取定义3D中的点和线的XYZL信息，并在MATLAB图形窗口中显示图像。

[XYZL\_DISPLAY\_OPENGL](https://people.sc.fsu.edu/~jburkardt/cpp_src/xyzl_display_opengl/xyzl_display_opengl.html)，一个C ++程序，读取定义3D中的点和线的XYZL信息，并使用OpenGL显示图像。

**源代码：**

* [xyz\_io.f90](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io.f90)，源代码。

**示例和测试：**

* [xyz\_io\_prb.f90](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io_prb.f90)，一个示例调用程序。
* [xyz\_io\_prb\_output.txt](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xyz_io_prb_output.txt)，输出文件。
* [helix.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/helix.xyz)，由示例程序创建的XYZ文件。
* [xy\_io\_prb\_02.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/xy_io_prb_02.xyz)，由示例程序创建的XYZ文件。
* [cube.xyz](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyz)，由示例程序创建的XYZ文件。
* [cube.xyzl](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyzl)，由示例程序创建的XYZL行文件。
* [cube.xyzf](https://people.sc.fsu.edu/~jburkardt/f_src/xyz_io/cube.xyzf)，由示例程序创建的XYZF行文件。

**例程列表：**

* **CH\_CAP**将单个字符大写。
* **CH\_EQI**是对两个字符进行不区分大小写的比较。
* **CH\_TO\_DIGIT**返回基数为10的整数值。
* **GET\_UNIT**返回一个免费的FORTRAN单元号。
* **S\_TO\_I4**从字符串中读取整数值。
* **S\_TO\_I4VEC**从字符串中读取整数向量。
* **S\_TO\_R8**从字符串中读取R8值。
* **S\_TO\_R8VEC**从字符串中读取R8VEC。
* **S\_WORD\_COUNT**计算字符串中“单词”的数量。
* **TIMESTAMP**将当前YMDHMS日期打印为时间戳。
* **TIMESTRING**将当前YMDHMS日期写入字符串。
* **WORD\_NEXT\_READ** “读取”字符串中的单词，一次一个。
* **XYZ\_DATA\_PRINT**打印XYZ文件的数据。
* **XYZ\_DATA\_READ**从XYZ文件中读取数据。
* **XYZ\_DATA\_WRITE**写入XYZ文件的数据。
* **XYZ\_EXAMPLE**计算示例XYZ数据集的点。
* **XYZ\_EXAMPLE\_SIZE调整**示例XYZ数据集的大小。
* **XYZ\_HEADER\_PRINT**打印XYZ文件的标题。
* **XYZ\_HEADER\_READ**确定XYZ文件中的数据对数。
* **XYZ\_HEADER\_WRITE**写入XYZ文件的标题。
* **XYZ\_READ**从XYZ文件中读取图形信息。
* **XYZ\_WRITE**写入XYZ文件。
* **XYZ\_WRITE\_TEST**测试XYZ写例程。
* **XYZF\_DATA\_PRINT**打印XYZF文件的数据。
* **XYZF\_DATA\_READ**读取XYZF文件中的数据。
* **XYZF\_DATA\_WRITE**写入XYZF文件的数据。
* **XYZF\_EXAMPLE**设置适合一对XYZ和XYZF文件的数据。
* **XYZF\_EXAMPLE\_SIZE调整XYZF\_EXAMPLE**创建的数据的大小。
* **XYZF\_HEADER\_PRINT**打印XYZF文件的标题。
* **XYZF\_HEADER\_READ**确定XYZF文件中的面部项目数。
* **XYZF\_HEADER\_WRITE**写入XYZF文件的标题。
* **XYZF\_WRITE**写入XYZF文件。
* **XYZL\_DATA\_PRINT**打印XYZL文件的数据。
* **XYZL\_DATA\_READ**读取XYZL文件中的数据。
* **XYZL\_DATA\_WRITE**写入XYZL文件的数据。
* **XYZL\_EXAMPLE**设置适合一对XYZ和XYZL文件的数据。
* **XYZL\_EXAMPLE\_SIZE调整XYZL\_EXAMPLE**创建的数据的大小。
* **XYZL\_HEADER\_PRINT**打印XYZL文件的标题。
* **XYZL\_HEADER\_READ**确定XYZL文件中的行项目数。
* **XYZL\_HEADER\_WRITE**写入XYZL文件的标题。
* **XYZL\_WRITE**写入XYZL文件。

您可以上一级到[FORTRAN90源代码](https://people.sc.fsu.edu/~jburkardt/f_src/f_src.html)。

*最后修订于2009年1月8日。*