UF_MTX2_copy (view source)

Defined in: uf_mtx.h

Overview
Copies the 2x2 matrix elements from the source matrix to the destination matrix

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_copy
(
    const double mtx_src [ 4 ] ,
    double mtx_dst [ 4 ]
)
```

const double	**mtx_src [4]**	Input	Source matrix
double	**mtx_dst [4]**	Output	Destination matrix mtx_dst = mtx_src

---

UF_MTX2_determinant (view source)

Defined in: uf_mtx.h

Overview
Calculates the determinant of a 2 x 2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_determinant
(
    const double mtx [ 4 ] ,
    double * determinant
)
```

const double	**mtx [4]**	Input	Matrix whose determinant in required
double *	**determinant**	Output	Matrix determinant

UF_MTX2_identity (view source)

Defined in: uf_mtx.h

Overview
Returns a 2 x 2 identity matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_identity
(
    double identity_mtx [ 4 ]
)
```

double	**identity_mtx [4]**	Output	Identity matrix

UF_MTX2_initialize (view source)

Defined in: uf_mtx.h

Overview
Returns a matrix formed from two input vectors.

Return
Return value:
0 = Success (vectors define a valid matrix)
1 = Matrix cannot be defined

Environment
Internal and External

Required License(s)
gateway

```
int UF_MTX2_initialize
(
    const double x_vec [ 2 ] ,
    const double y_vec [ 2 ] ,
    double mtx [ 4 ]
)
```

const double	**x_vec [2]**	Input	Vector for the X-direction of matrix

const double	**y_vec [2]**	Input	Vector for the Y-direction of matrix
double	**mtx [4]**	Output	Matrix

UF_MTX2_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 2x2 matrix product from two input matrices.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_multiply
(
    const double mtx1 [ 4 ] ,
    const double mtx2 [ 4 ] ,
    double mtx_product [ 4 ]
)
```

const double	**mtx1 [4]**	Input	Matrix #1
const double	**mtx2 [4]**	Input	Matrix #2
double	**mtx_product [4]**	Output	Matrix product mtx_product = mtx1 X mtx2

UF_MTX2_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a 2x2 matrix product by transposing matrix #1 before performing the multiplication.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_multiply_t
(
    const double mtx1 [ 4 ] ,
    const double mtx2 [ 4 ] ,
    double mtx_product [ 4 ]
)
```

const double	**mtx1 [4]**	Input	Matrix #1 gets transposed before the multiplication.
const double	**mtx2 [4]**	Input	Matrix #2
double	**mtx_product [4]**	Output	Matrix product mtx_product = Transpose of mtx1 X mtx2

UF_MTX2_transpose (view source)

Defined in: uf_mtx.h

Overview
Returns the transpose of a 2x2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_transpose
(
    const double mtx [ 4 ] ,
    double transpose_mtx [ 4 ]
)
```

const double	**mtx [4]**	Input	Matrix to transpose
double	**transpose_mtx [4]**	Output	Transpose of the input matrix

---

UF_MTX2_vec_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 2D vector which is the product of a 2D vector and a 2x2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_vec_multiply
(
    const double vec [ 2 ] ,
    const double mtx [ 4 ] ,
    double vec_product [ 2 ]
)
```

const double	**vec [2]**	Input	Vector to multiply
const double	**mtx [4]**	Input	Matrix to multiply
double	**vec_product [2]**	Output	Product (a vector) vec_product = vec X mtx

---

UF_MTX2_vec_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a 2D vector which is the product of a 2D vector and a transposed 2x2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_vec_multiply_t
(
    const double vec [ 2 ] ,
    const double mtx [ 4 ] ,
    double vec_product [ 2 ]
)
```

const double	**vec [2]**	Input	Vector to multiply
const double	**mtx [4]**	Input	Matrix to transpose and multiply
double	**vec_product [2]**	Output	Product (a vector) vec_product = vec X transpose of mtx

UF_MTX2_x_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the X-direction vector of a 2x2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_x_vec
(
    const double mtx [ 4 ] ,
    double x_vec [ 2 ]
)
```

const double	**mtx [4]**	Input	Matrix whose X-direction is required
double	**x_vec [2]**	Output	X-direction vector of the matrix

UF_MTX2_y_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the Y-direction vector of a 2x2 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX2_y_vec
(
    const double mtx [ 4 ] ,
    double y_vec [ 2 ]
)
```

const double	**mtx [4]**	Input	Matrix whose Y-direction is required

double	**y_vec [2]**	Output	Y-direction vector of the matrix

UF_MTX3_copy (view source)

Defined in: uf_mtx.h

Overview
Copies the matrix elements from a source 3x3 matrix to a destination 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_copy
(
    const double mtx_src [ 9 ] ,
    double mtx_dst [ 9 ]
)
```

const double	**mtx_src [9]**	Input	Source matrix
double	**mtx_dst [9]**	Output	Destination matrix

UF_MTX3_determinant (view source)

Defined in: uf_mtx.h

Overview
Calculates the determinant of a 3 x 3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_determinant
(
    const double mtx [ 9 ] ,
    double * determinant
)
```

const double	**mtx [9]**	Input	Matrix whose determinant in required
double *	**determinant**	Output	Matrix determinant

UF_MTX3_identity (view source)

Defined in: uf_mtx.h

Overview
Returns a 3 x 3 identity matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_identity
(
    double identity_mtx [ 9 ]
)
```

double	**identity_mtx [9]**	Output	Identity Matrix

UF_MTX3_initialize (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix formed from two input 3D vectors. The two input vectors are normalized and the y-direction vector is made orthogonal to the x-direction vector before taking the cross product (x_vec X y_vec) to generate the z-direction vector.

Return
Return value:
0 = Success (vectors define a valid matrix)
1 = Matrix cannot be defined

Environment
Internal and External

Required License(s)
gateway

```
int UF_MTX3_initialize
(
    const double x_vec [ 3 ] ,
    const double y_vec [ 3 ] ,
    double mtx [ 9 ]
)
```

const double	**x_vec [3]**	Input	Vector for the X-direction of matrix
const double	**y_vec [3]**	Input	Vector for the Y-direction of matrix
double	**mtx [9]**	Output	Matrix

UF_MTX3_initialize_x (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix with the given X-direction vector and having arbitrary Y- and Z-direction vectors.

Return
Returns 0 if the input vector is nonzero; returns 1 otherwise.

Environment
Internal and External

Required License(s)
gateway

```
int UF_MTX3_initialize_x
(
    const double x_vec [ 3 ] ,
    double mtx [ 9 ]
)
```

const double	**x_vec [3]**	Input	Vector for the X-direction of matrix
double	**mtx [9]**	Output	Matrix (3x3)

UF_MTX3_initialize_z (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix with the given Z-direction vector and having arbitrary X- and Y-direction vectors.

Return

Returns 0 if the input vector is nonzero;
returns 1 otherwise.

Environment

Internal and External

Required License(s)

gateway

```
int UF_MTX3_initialize_z
(
    const double z_vec [ 3 ] ,
    double mtx [ 9 ]
)
```

const double	**z_vec [3]**	Input	Vector for the Z-direction of matrix
double	**mtx [9]**	Output	Matrix (3x3)

UF_MTX3_mtx4 (view source)

Defined in: uf_mtx.h

Overview

Converts a 3D matrix to a 4D matrix with a scale of 1.0 and a zero translation vector.

Return

void

Environment

Internal and External

Required License(s)

gateway

```
void UF_MTX3_mtx4
(
    const double mtx_3D [ 9 ] ,
    double mtx_4D [ 16 ]
)
```

const double	**mtx_3D [9]**	Input	3D matrix
double	**mtx_4D [16]**	Output	4D matrix

UF_MTX3_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix product from two input matrices.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_multiply
(
    const double mtx1 [ 9 ] ,
    const double mtx2 [ 9 ] ,
    double mtx_product [ 9 ]
)
```

const double	**mtx1 [9]**	Input	Matrix #1
const double	**mtx2 [9]**	Input	Matrix #2
double	**mtx_product [9]**	Output	Matrix product mtx_product = mtx1 X mtx2

UF_MTX3_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix product by transposing the first matrix before performing the multiplication.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_multiply_t
(
    const double mtx1 [ 9 ] ,
    const double mtx2 [ 9 ] ,
    double mtx_product [ 9 ]
)
```

const double	**mtx1 [9]**	Input	Matrix #1 gets transposed before the multiplication.
const double	**mtx2 [9]**	Input	Matrix #2

double	**mtx_product [9]**	Output	Matrix product mtx_product = trns(mtx1) X mtx2

UF_MTX3_ortho_normalize (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 matrix whose direction vectors are orthogonal and of unit length.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_ortho_normalize
(
    double mtx [ 9 ]
)
```

double	**mtx [9]**	Input / Output	Matrix to be ortho-normalized. (Input) Ortho-normalized matrix. (Output)

UF_MTX3_rotate_about_axis (view source)

Defined in: uf_mtx.h

Overview
Returns a 3x3 rotation matrix about an axis and through a specified angle of rotation.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_rotate_about_axis
(
    const double rotation_axis [ 3 ] ,
    double rotation_angle,
    double mtx [ 9 ]
```

)

const double	**rotation_axis [3]**	Input	Vector of the rotation axis
double	**rotation_angle**	Input	Angle of the rotation (in radians)
double	**mtx [9]**	Output	Rotation Matrix

UF_MTX3_transpose (view source)

Defined in: uf_mtx.h

Overview
Returns the transpose of a 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

void UF_MTX3_transpose
(
const double mtx [9] ,
double transpose_mtx [9]
)

const double	**mtx [9]**	Input	Matrix to transpose
double	**transpose_mtx [9]**	Output	Transposed matrix transpose_mtx = trns(mtx)

UF_MTX3_vec_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a vector which is the product of a 3D vector and a 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_vec_multiply
(
    const double vec [ 3 ] ,
    const double mtx [ 9 ] ,
    double vec_product [ 3 ]
)
```

const double	**vec [3]**	Input	Vector to multiply
const double	**mtx [9]**	Input	Matrix to multiply
double	**vec_product [3]**	Output	Product (a vector) vec_product = vec X mtx

---

UF_MTX3_vec_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a vector which is the product of a 3D vector and a transposed 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_vec_multiply_t
(
    const double vec [ 3 ] ,
    const double mtx [ 9 ] ,
    double vec_product [ 3 ]
)
```

const double	**vec [3]**	Input	Vector to multiply
const double	**mtx [9]**	Input	Matrix to transpose and multiply
double	**vec_product [3]**	Output	Product (a vector) vec_product = vec X trns(mtx)

UF_MTX3_x_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the X-direction vector of a matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_x_vec
(
    const double mtx [ 9 ] ,
    double x_vec [ 3 ]
)
```

const double	**mtx [9]**	Input	3x3 Matrix whose X-direction is required
double	**x_vec [3]**	Output	X-direction vector of the matrix

UF_MTX3_y_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the Y-direction vector of a 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_y_vec
(
    const double mtx [ 9 ] ,
    double y_vec [ 3 ]
)
```

const double	**mtx [9]**	Input	Matrix whose Y-direction is required
double	**y_vec [3]**	Output	Y-direction vector of the matrix

UF_MTX3_z_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the Z-direction vector of a 3x3 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX3_z_vec
(
    const double mtx [ 9 ] ,
    double z_vec [ 3 ]
)
```

const double	**mtx [9]**	Input	Matrix whose Z-direction is required
double	**z_vec [3]**	Output	Z-direction vector of the matrix

UF_MTX4_ask_rotation (view source)

Defined in: uf_mtx.h

Overview
Returns the 3x3 rotation matrix of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_ask_rotation
(
    const double mtx_4D [ 16 ] ,
    double mtx_3D [ 9 ]
)
```

const double	**mtx_4D [16]**	Input	4x4 matrix whose rotation is required
double	**mtx_3D [9]**	Output	3x3 rotation matrix of the 4x4 matrix

UF_MTX4_ask_scale (view source)

Defined in: uf_mtx.h

Overview
Returns the scale factor of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_ask_scale
(
    const double mtx [ 16 ] ,
    double * scale
)
```

const double	**mtx [16]**	Input	Matrix whose scale is required.
double *	**scale**	Output	Scale factor of the matrix

UF_MTX4_ask_translation (view source)

Defined in: uf_mtx.h

Overview
Returns the translation vector of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_ask_translation
(
    const double mtx [ 16 ] ,
    double translate_vec [ 3 ]
)
```

const double	**mtx [16]**	Input	Matrix whose translation is required.
double	**translate_vec [3]**	Output	Translation vector of the matrix

UF_MTX4_copy (view source)

Defined in: uf_mtx.h

Overview
Copies 4x4 matrix elements from the source matrix to the destination matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_copy
(
    const double mtx_src [ 16 ] ,
    double mtx_dst [ 16 ]
)
```

const double	**mtx_src [16]**	Input	Source matrix
double	**mtx_dst [16]**	Output	Destination matrix mtx_dst = mtx_src

UF_MTX4_csys_to_csys (view source)

Defined in: uf_mtx.h

Overview
Returns the matrix which can be used to map from one csys to another.

Environment
Internal and External

See Also
Refer to example

History
Originally released in V16.0

Required License(s)
gateway

```
int UF_MTX4_csys_to_csys
(
    const double from_origin [ 3 ] ,
    const double from_x_axis [ 3 ] ,
    const double from_y_axis [ 3 ] ,
    const double to_origin [ 3 ] ,
    const double to_x_axis [ 3 ] ,
    const double to_y_axis [ 3 ] ,
    double mtx [ 16 ]
)
```

const double	**from_origin [3]**	Input	origin of csys to map from
const double	**from_x_axis [3]**	Input	x axis of csys to map from
const double	**from_y_axis [3]**	Input	y axis of csys to map from
const double	**to_origin [3]**	Input	origin of csys to map to
const double	**to_x_axis [3]**	Input	x axis of csys to map to
const double	**to_y_axis [3]**	Input	y axis of csys to map to
double	**mtx [16]**	Output	Returned matrix that can be used to tranform objects

UF_MTX4_edit_rotation (view source)

Defined in: uf_mtx.h

Overview
Edits the 3x3 rotation matrix of a 4 x 4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_edit_rotation
(
    double mtx_4D [ 16 ] ,
    const double mtx_3D [ 9 ]
)
```

double	**mtx_4D [16]**	Input / Output	4x4 matrix whose rotation is to be edited. (Input) 4x4 with an edited 3x3 rotation matrix. (Output)
const double	**mtx_3D [9]**	Input	3x3 rotation matrix to use as replacement in 4x4 matrix.

UF_MTX4_edit_scale (view source)

Defined in: uf_mtx.h

Overview
Edits the scale factor of a 4 x 4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_edit_scale
(
    double mtx [ 16 ] ,
    double scale
)
```

double	**mtx [16]**	Input / Output	4x4 matrix whose scale is to be edited. (Input) 4x4 with an edited scale factor. (Output)
double	**scale**	Input	Scale factor to use as replacement in 4x4 matrix.

UF_MTX4_edit_translation (view source)

Defined in: uf_mtx.h

Overview
Edits the translation vector of a 4 x 4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_edit_translation
(
    double mtx [ 16 ] ,
    const double translate_vec [ 3 ]
)
```

double	**mtx [16]**	Input / Output	4x4 matrix whose translation is to be edited. (Input) 4x4 with an edited translation vector. (Output)

const double	**translate_vec [3]**	Input	Translation vector to use as replacement in 4x4 matrix.

UF_MTX4_identity (view source)

Defined in: uf_mtx.h

Overview
Returns a 4 x 4 identity matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_identity
(
    double identity_mtx [ 16 ]
)
```

double	**identity_mtx [16]**	Output	Identity Matrix

UF_MTX4_initialize (view source)

Defined in: uf_mtx.h

Overview
Returns the 4x4 matrix formed from a 3x3 rotation matrix, a 3D translation vector, and a scale factor.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_initialize
(
    double scale,
    const double translation_vec [ 3 ] ,
    const double mtx_3D [ 9 ] ,
    double mtx_4D [ 16 ]
)
```

double	**scale**	Input	Scale factor
const double	**translation_vec [3]**	Input	Translation vector
const double	**mtx_3D [9]**	Input	3x3 rotation matrix
double	**mtx_4D [16]**	Output	4x4 matrix

UF_MTX4_invert (view source)

Defined in: uf_mtx.h

Overview
Returns the matrix which is the invert of the input one

Return
Return value:
0 = Success (inverted matrix created)
n = Matrix not defined

Environment
Internal and External

History
Originally released in V18.0

Required License(s)
gateway

```
int UF_MTX4_invert
(
    const double mtx_in [ 16 ] ,
    double mtx_out [ 16 ]
)
```

const double	**mtx_in [16]**	Input	Input matrix
double	**mtx_out [16]**	Output	Returned inverted matrix

UF_MTX4_mirror (view source)

Defined in: uf_mtx.h

Overview
Returns the matrix which can be used to mirror about a plane

Environment
Internal and External

See Also

Refer to example

History
Originally released in V16.0

Required License(s)
gateway

```
int UF_MTX4_mirror
(
    const double origin [ 3 ] ,
    const double normal [ 3 ] ,
    double mtx [ 16 ]
)
```

const double	**origin [3]**	Input	The origin of the plane.
const double	**normal [3]**	Input	The plane normal
double	**mtx [16]**	Output	Returned matrix that can be used to tranform objects

---

UF_MTX4_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 4x4 matrix product from two input matrices.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_multiply
(
    const double mtx1 [ 16 ] ,
    const double mtx2 [ 16 ] ,
    double mtx_product [ 16 ]
)
```

const double	**mtx1 [16]**	Input	Matrix #1
const double	**mtx2 [16]**	Input	Matrix #2
double	**mtx_product [16]**	Output	Matrix product mtx_product = mtx1 X mtx2

UF_MTX4_multiply_t (view source)

Defined in: uf_mtx.h

Overview

Returns a 4x4 matrix product by transposing the first matrix before performing the multiplication.

Return

void

Environment

Internal and External

Required License(s)

gateway

```
void UF_MTX4_multiply_t
(
    const double mtx1 [ 16 ] ,
    const double mtx2 [ 16 ] ,
    double mtx_product [ 16 ]
)
```

const double	**mtx1 [16]**	Input	Matrix #1 gets transposed before the multiplication.
const double	**mtx2 [16]**	Input	Matrix #2
double	**mtx_product [16]**	Output	Matrix product mtx_product = trns(mtx1) X mtx2

UF_MTX4_ortho_normalize (view source)

Defined in: uf_mtx.h

Overview

Returns a 4x4 matrix whose direction vectors are orthogonal and of unit length.

Return

void

Environment

Internal and External

Required License(s)

gateway

```
void UF_MTX4_ortho_normalize
(
    double mtx [ 16 ]
)
```

double	**mtx [16]**	Input / Output	Matrix to be ortho-normalized. (Input) Ortho-normalized matrix. (Output)

UF_MTX4_rotation (view source)

Defined in: uf_mtx.h

Overview
Returns the matrix which can be used to rotate about a point

Environment
Internal and External

See Also
Refer to example

History
Originally released in V16.0

Required License(s)
gateway

```
int UF_MTX4_rotation
(
    const double rotation_point [ 3 ] ,
    const double rotation_axis [ 3 ] ,
    const double angle,
    double mtx [ 16 ]
)
```

const double	**rotation_point [3]**	Input	Point about which the rotation is to be performed.
const double	**rotation_axis [3]**	Input	Axis about which rotation to occur.
const double	**angle**	Input	rotation angle in degrees
double	**mtx [16]**	Output	returned matrix that can be used to tranform objects

---

UF_MTX4_scaling (view source)

Defined in: uf_mtx.h

Overview
Returns the matrix using scaling and invariant point information.

Environment
Internal and External

See Also

Refer to example

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_MTX4_scaling
(
    const double invariant_point [ 3 ] ,
    const double scale [ 3 ] ,
    double mtx [ 16 ]
)
```

const double	**invariant_point [3]**	Input	Point which will be invariant to the scaling, in other words the center point of the scale operation.
const double	**scale [3]**	Input	scaling in x, y, z directions
double	**mtx [16]**	Output	returned matrix that can be used to tranform objects

---

UF_MTX4_transpose (view source)

Defined in: uf_mtx.h

Overview

Returns the transpose of a 4x4 matrix.

Return

void

Environment

Internal and External

Required License(s)

gateway

```
void UF_MTX4_transpose
(
    const double mtx [ 16 ] ,
    double transpose_mtx [ 16 ]
)
```

const double	**mtx [16]**	Input	Matrix to transpose
double	**transpose_mtx [16]**	Output	Transposed matrix transpose_mtx = trns(mtx)

UF_MTX4_vec3_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 3D vector which is the product of a 3D vector and a 4x4 matrix. The 3D vector is treated as a 4D vector with a weight of 1.0.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_vec3_multiply
(
    const double vec [ 3 ] ,
    const double mtx [ 16 ] ,
    double vec_product [ 3 ]
)
```

const double	**vec [3]**	Input	Vector to multiply
const double	**mtx [16]**	Input	Matrix to multiply
double	**vec_product [3]**	Output	Product (a vector) vec_product = vec X mtx

---

UF_MTX4_vec3_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a 3D vector which is the product of a 3D vector and the transpose of a 4x4 matrix. During the multiplication, the 3D vector is treated as a 4D vector with a weight of 1.0.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_vec3_multiply_t
(
    const double vec [ 3 ] ,
    const double mtx [ 16 ] ,
    double vec_product [ 3 ]
)
```

const double	**vec [3]**	Input	Vector to multiply
const double	**mtx [16]**	Input	Matrix to multiply
double	**vec_product [3]**	Output	Product (a vector) vec_product = vec X trns(mtx)

UF_MTX4_vec_multiply (view source)

Defined in: uf_mtx.h

Overview
Returns a 4D vector which is the product of a 4D vector and a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_vec_multiply
(
    const double vec [ 4 ] ,
    const double mtx [ 16 ] ,
    double vec_product [ 4 ]
)
```

const double	**vec [4]**	Input	Vector to multiply
const double	**mtx [16]**	Input	Matrix to multiply
double	**vec_product [4]**	Output	Product (a vector) vec_product = vec X mtx

---

UF_MTX4_vec_multiply_t (view source)

Defined in: uf_mtx.h

Overview
Returns a vector which is the product of a 4D vector and a transposed matrix.

Return

void

Environment
Internal and External

Required License(s)
gateway

void UF_MTX4_vec_multiply_t
(
const double vec [4] ,
const double mtx [16] ,
double vec_product [4]
)

const double	**vec [4]**	Input	Vector to multiply
const double	**mtx [16]**	Input	Matrix to transpose and multiply
double	**vec_product [4]**	Output	Product (a vector) vec_product = vec X trns(mtx)

UF_MTX4_x_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the X-direction vector of the 3x3 rotation of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

void UF_MTX4_x_vec
(
const double mtx [16] ,
double x_vec [3]
)

const double	**mtx [16]**	Input	4x4 Matrix whose X-direction is required
double	**x_vec [3]**	Output	X-direction vector of the matrix

UF_MTX4_y_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the Y-direction vector of the 3x3 rotation of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_y_vec
(
    const double mtx [ 16 ] ,
    double y_vec [ 3 ]
)
```

const double	**mtx [16]**	Input	4x4 matrix whose Y-direction is required
double	**y_vec [3]**	Output	Y-direction vector of the matrix

UF_MTX4_z_vec (view source)

Defined in: uf_mtx.h

Overview
Returns the Z-direction vector of the 3x3 rotation of a 4x4 matrix.

Return
void

Environment
Internal and External

Required License(s)
gateway

```
void UF_MTX4_z_vec
(
    const double mtx [ 16 ] ,
    double z_vec [ 3 ]
)
```

const double	**mtx [16]**	Input	Matrix whose Z-direction is required
double	**z_vec [3]**	Output	Z-direction vector of the matrix
