…

// example for M = 6, N = 3

size\_t dim\_count = 2;

size\_t dims[] = {6,3};

struct matrix \*A = matrix\_init(dim\_count, dims);

printf("A: \n");

matrix\_print\_metadata(A);

matrix\_print(A);

// or just use matrix\_print\_all(A) instead of previous 3 lines

matrix\_destroy(A);

…

struct matrix \*x = matrix\_sin\_linspace(0, MATRIX\_PI, 10);

struct matrix \*y = matrix\_cos\_range(0, 30, 5);

struct matrix \*A = matrix\_crossproduct(y, x);

matrix\_print\_all(A);

matrix\_destroy\_batch(3, x, y, A);

size\_t pos[2];

int i, j;

for (i=1; i<=6; i++) {

for (j=1; j<=3; j++) {

pos[0] = i;

pos[1] = j;

matrix\_set\_element(A, pos, 2\*i+j\*j);

}

}

struct matrix \*t = matrix\_linspace(0, MATRIX\_PI, 10);

struct matrix \*A = matrix\_sin(t);

matrix\_destroy\_batch(2, t, A);

struct matrix \*AT = matrix\_transpose(A);

// struct matrix \*y = matrix\_cos\_range(0, 30, 5) to a 6x3 matrix

size\_t new\_dim\_count = 2;

size\_t new\_dims[] = {6,3}

struct matrix \*y1 = matrix\_reshape(A, new\_dim\_count, new\_dims);

N

2

1

N

2

1