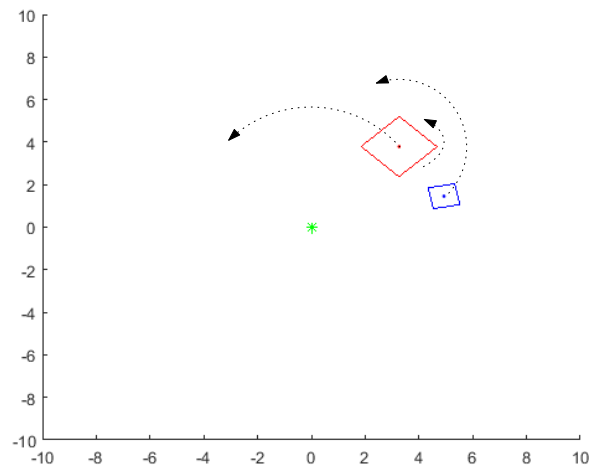


Basics of Computer Graphics: Exercise Set 1

Goal: The aim of this exercise is to implement an animation of Sun, Earth and Moon.



Task 1 Implement the following two auxiliary methods

(1) $p_{\text{new}} = \text{rotate_around_point}(p_{\text{old}}, a, b, \phi)$

Input:

- a point p_{old} represented via homogeneous coordinates
- numbers a, b and an angle ϕ (in radians)

Output: homogeneous coordinates of the point obtained by rotating p_{old} around the point $(a; b)$ by ϕ

Test cases:

- `rotate_around_point([2;5;1], 6, 9, pi/2)` gives `[10;5;1]`
- `rotate_around_point([-2;8;1], 5, 7, pi)` gives `[12; 6; 1]`

(2) $[p1_{\text{new}}, p2_{\text{new}}, p3_{\text{new}}, p4_{\text{new}}] =$

`rotate_tuple(p1old, p2old, p3old, p4old, a, b, ϕ),`

which applies the rotation described in (1) to each of the 4 input-points separately.

Task 2 Complete the program `earth_sun1` (which can be found via **Material/Part2 - Computer Graphics/Code on olat**) to run an animation of an Earth (modeled by a point) orbiting a Sun (modeled by an asterisk in the origin).

Task 3 Adapt the program `earth_sun1` such that the Earth is represented by a square with a mark in its center.

Name your program `earth_sun2`.

Task 4 Modify `earth_sun2` such that the Earth also spins on its center.

Name the resulting program `earth_sun3`.

Task 5 (optional) Adapt `earth_sun3` such that it also contains a Moon (represented by a square with a mark in its center) which

- orbits the Earth, and
- spins on its center.

Name the resulting program `earth_sun_moon`.