```
// longitude/latitude of location
lon=11.42*%CL_deg2rad;
lat=48.15*%CL_deg2rad;
alt=531;

// scal = reference date (string, calendar)
scal = [];
desc_param=list(..
 CL_defParam("Longitude of location", lon, units=["rad", "deg"],..
               valid='$x>=0 & $x <=360'), ..
 CL_defParam("Latitude of location", lat, units=["rad", "deg"],..
               valid='$x>=-90 & $x <=90'), ..
 CL_defParam("Reference time (calendar format, TREF)",scal,..
               typ="cal"));

[lon,lat,scal]=CL_inputParam(desc_param);
loc=[lon;lat;0];
t0=CL_dat_cal2cjd(CL_dat_str2cal(scal));

// Position of the sun in ECI
pos_sun=CL_eph_sun(t0);

// Position of the venus in ICRS
pos_ICRS=CL_eph_de405("Venus",t0,"Earth");
// Position of the venus in ECI
pos_venus=CL_fr_convert("ICRS","ECI",t0,pos_ICRS);

// Real position of the observer in ECF
// Geodetic to cartesian coordinates
pos_ECF=CL_co_ell2car(loc);

// Real position of the observer in ECI
pos_obs=CL_fr_convert("ECF","ECI",t0,pos_ECF);

// unit vectors for directions
uv_pos_obs=pos_obs/CL_norm(pos_obs);
uv_obs_ve=(pos_obs-pos_venus)/CL_norm(pos_obs-pos_venus);
uv_su_ve=(pos_sun-pos_venus)/CL_norm(pos_sun-pos_venus);

// Observer local frame - z-axis from observer to venus
M_obs=CL_rot_defFrameVec(uv_obs_ve,uv_pos_obs,3,2);

// Frame for illuminating the venus - z-axis from
// venus to sun
M=CL_rot_defFrameVec(uv_su_ve,uv_su_ve,3,3);

RotM=M_obs*M';
disp(M_obs);
```