

## VJEŽBA 1: OPIS POLOŽAJA TIJELA U PROSTORU

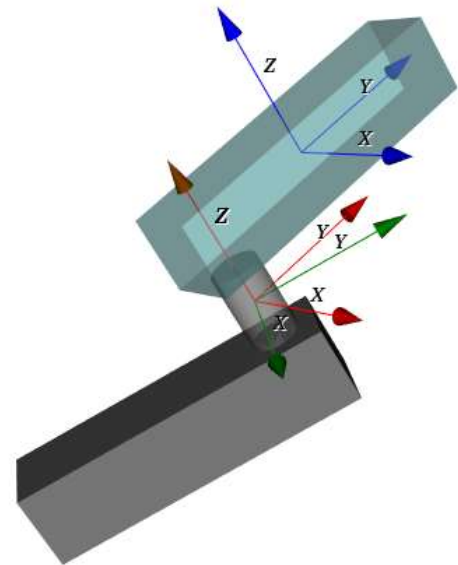
**I. Cilj vježbe:** Naučiti kako se koriste rotacijske matrice i matrice homogene transformacije za opis položaja tijela u prostoru.

### II. Opis vježbe:

Na slici 1 prikazana su dva kvadra, koji predstavljaju članke robotskog manipulatora povezane rotacijskim zglobovima prikazanim valjkom. Članak 1 prikazan je sivom, a članak 2 tirkiznom bojom. Zglob prikazan sivim cilindrom predstavlja dio članka 1. Svaki članak ima pridružen koordinatni sustav s ishodištem u centroidu kvadra i osima poravnatim sa stranicama kvadra. Koordinatni sustav pridružen članku 2 prikazan je plavom bojom. Zglob ima pridružen koordinatni sustav s ishodištem u centroidu valjka te z-osi paralelnom s osi valjka. Koordinatni sustav valjka prikazan je na slici 1 zelenom bojom. Članak 2 ima također pridružen i koordinatni sustav koji ga povezuje s člankom 1 i koji je na slici 1 prikazan crvenom bojom. Osi svih koordinatnih sustava pridruženih nekom članku su paralelne.

Ovaj primjer treba poslužiti kao predložak za rješenje sljedećeg zadatka. Zadatak je izraditi aplikaciju koja prikazuje 3D model troosnog robotskog manipulatora (robot s tri članka povezana s tri zgloba) korištenjem matrica homogene transformacije.

Program treba izraditi korištenjem biblioteke Visualization Toolkit (VTK) i klase *Body* i *Display* izrađenih za potrebe ove laboratorijske vježbe.



Sl. 1 Dva članka robotskog manipulatora povezana rotacijskim zglobovima.

**Klasa *Body*** namijenjena je prikazu geometrijskih tijela u 3D prostoru. Funkcija *CreateBox* ove klase služi za stvaranje kvadara.

```
Body.CreateBox(a, b, c, red, green, blue)
```

Argumenti ove funkcije su duljine stranica kvadra  $a$ ,  $b$  i  $c$  te boja kvadra definirana crvenom, zelenom i plavom komponentom *red*, *green* i *blue*.

Klasa *Body* također posjeduje i funkciju *CreateCylinder*, koja služi za stvaranje valjaka.

```
Body.CreateCylinder(r, h, res, red, green, blue)
```

Argumenti ove funkcije su polumjer valjka  $r$ , njegova visina  $h$ , preciznost prikaza valjka  $res$  te njegova boja definirana kao i kod funkcije *CreateBox*. Valjak je zapravo aproksimiran pravilnom višestranom prizmom, gdje je broj stranica prizme definiran parametrom  $res$ .

Položaj tijela prikazanog pomoću klase *Body* u prostoru definira se funkcijom *Transform*.

```
Body.Transform(T)
```

Argument ove funkcije je matrica homogene transformacije  $T$ , koja je zadana kao polje od 16 brojeva, koji predstavljaju elemente matrice zapisane redak po redak.

Scena, koja se sastoji od više geometrijskih tijela, također se prikazuje klasom *Body* tako da se pojedinačno geometrijsko tijelo dodaje na scenu funkcijom *AddPart*.

```
Body.AddPart(part)
```

Argument ove funkcije *part* je tijelo koje se želi dodati na scenu.

Scena se prikazuje u prozoru, koji se realizira klasom *Display*, pomoću funkcije *AddToDisplay*.

```
Body.AddToDisplay(window)
```

Argument ove funkcije je instanca klase *Display*, koja predstavlja prozor na kojemu se želi prikazati scena.

**Klasa *Display*** služi za stvaranje prozora na kojemu se prikazuje scena stvorena pomoću klase *Body*. Prozor se stvara pomoću konstruktora.

Display(w, h, name, red, green, blue)

Argumenti ove funkcije su širina  $w$  i visina  $h$  prozora, njegovo ime  $name$  te boja pozadine definirana kao i kod funkcije *CreateBox*.

Vizualizacija scene pokreće se funkcijom *Run*.

Display.Run()

**Množenje matrica homogene transformacije u VTK** se može realizirati pomoću funkcije *Multiply4x4* klase *vtkMatrix4x4*. Primjer množenja dvije matrice dan je u nastavku.

```
T10 = [
    0, 0, 1, 0.0,
    1, 0, 0, 0.0,
    0, 1, 0, 0.125,
    0, 0, 0, 1.0
]
```

```
T21 = [
    1, 0, 0, 0,
    0, 1, 0, 0.15,
    0, 0, 1, 0,
    0, 0, 0, 1
]
```

```
T20 = [0] * 16
```

```
vtkMatrix4x4.Multiply4x4(T10, T21, T20)
```

Ovaj primjer demonstrira množenje dviju matrica homogene transformacije  $T10$  i  $T21$ , čiji se rezultat sprema u matricu  $T20$ .

### III. Rad na vježbi:

Izraditi konzolnu aplikaciju koja prikazuje troosni robotski manipulator u položaju definiranom od strane korisnika. Robot se treba sastojati od baze prikazane kvadrom, tri članka prikazana kvadrima, koji su međusobno povezani rotacijskim zglobovima prikazanim valjcima, kao na slici 1. Korisnik definira položaj robota zadavanjem kutova njegova tri zgloba.