HOMEWORK ASSIGNMENT - 2

Problem 1:

a) The rotation matrix relating local robot velocities with respect to global coordinates is:

$$R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

such that $\dot{\xi}_1 = R\,\dot{\xi}_R$

where $\dot{\xi}_1$ is the velocity matrix of the robot in the world frame and $\dot{\xi}_R$ is the velocity matrix of the robot in it's own frame.

Therefore, $$\dot{\xi}_1 = \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix}, \quad \dot{\xi}_R = \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = \begin{bmatrix} u \\ v \\ \omega \end{bmatrix}$$

b) Given: $u = 15$ cm/s $(\dot{x}_R)$
$v = 0$ cm/s $(\dot{y}_R)$
$\omega = 0$ rad/s $(\dot{\theta}_R)$
$\theta = \pi/6$ rad.

$\therefore$ Robot velocities in the global coordinate frame are:

$$\dot{\xi}_1 = R\,\dot{\xi}_R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix}$$

$$= \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \\ 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 15 \\ 0 \\ 0 \end{bmatrix}$$

$$\therefore \dot{\xi}_1 = \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix} = \begin{bmatrix} 15\sqrt{3}/2 \\ 15/2 \\ 0 \end{bmatrix} = \begin{bmatrix} 13 \\ 7.5 \\ 0 \end{bmatrix} \Rightarrow \begin{aligned} \dot{x}_o &= 13 \text{ cm/s} \\ \dot{y}_o &= 7.5 \text{ cm/s} \\ \dot{\theta}_o &= 0 \text{ rad/s} \end{aligned}$$

c) Given: $u = 0$ cm/s $(\dot{x}_R)$
$v = 0$ cm/s $(\dot{y}_R)$
$\omega = 2$ rad/s $(\dot{\theta}_R)$
$\theta = \pi/3$ rad.

$\therefore$ Robot velocities in the global coordinate frame are:

$$\dot{\xi}_I = R\,\dot{\xi}_R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix}$$

$$= \begin{bmatrix} 1/2 & -\sqrt{3}/2 & 0 \\ \sqrt{3}/2 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

$$\therefore \dot{\xi}_I = \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \Rightarrow \begin{aligned} \dot{x}_o &= 0 \text{ cm/s} \\ \dot{y}_o &= 0 \text{ cm/s} \\ \dot{\theta}_o &= 2 \text{ rad/s} \end{aligned}$$

d) Given: $u = 25$ cm/s $(\dot{x}_R)$
$v = 0$ cm/s $(\dot{y}_R)$
$\omega = 4$ rad/s $(\dot{\theta}_R)$
$\theta = \pi/4$ rad.

$\therefore$ Robot velocities in the global coordinate frame are:

$$\dot{\xi}_I = R\,\dot{\xi}_R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix}$$

$$= \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 25 \\ 0 \\ 4 \end{bmatrix}$$

$$\therefore \dot{\xi}_I = \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{\theta}_o \end{bmatrix} = \begin{bmatrix} 25/\sqrt{2} \\ 25/\sqrt{2} \\ 4 \end{bmatrix} \Rightarrow \begin{aligned} \dot{x}_o &= 25/\sqrt{2} \\ \dot{y}_o &= 25/\sqrt{2} \\ \dot{\theta}_o &= 4 \end{aligned}$$

e) Given: $\dot{x}_0 = 10$ cm/s

$\dot{y}_0 = 10$ cm/s

$\dot{\theta}_0 = 0$ rad./s

$\theta = -3\pi/4$ rad.

$\therefore$ the robot velocities in local coordinates from its velocities in the global ~~glo~~ coordinate frame,

(From $\dot{\xi}_1 = R\dot{\xi}_R$)

$$\dot{\xi}_R = R^{-1}\dot{\xi}_1$$

$$R^{-1} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\therefore \dot{\xi}_R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_0 \\ \dot{y}_0 \\ \dot{\theta}_0 \end{bmatrix}$$

$$= \begin{bmatrix} -1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 10 \\ 10 \\ 0 \end{bmatrix}$$

$$\therefore \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = \begin{bmatrix} -10\sqrt{2} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -14.14 \\ 0 \\ 0 \end{bmatrix}$$

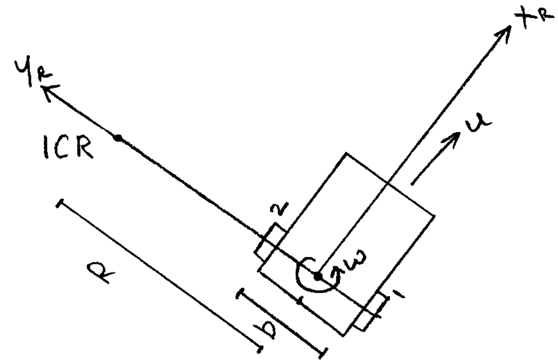
$\therefore \dot{x}_R = -14.4$ cm/s (u)

$\dot{y}_R = 0$ cm/s (v)

$\dot{\theta}_R = 0$ rad./s (w)

Problem 2:

a)



$\dot{\phi}, r$

$r_1 = r_2 = r$

Consider the above additions to the diagram in Problem 1. Here, ICR is the instantaneous centre of rotation of the robot such that it is located at a distance of R from the midpoint between the two wheels of the robot.

For any rate of rotation, ω, it remains same for both the wheels about the ICR. Therefore,

for wheel 1, $\omega(R + b/2) = V_1 = \dot{\phi}_1 r$ —— (1)

for wheel 2, $\omega(R - b/2) = V_2 = \dot{\phi}_2 r$ —— (2)

Solving the equations (1) and (2), ~~(1)+(2)~~ we get

$$R = \frac{br}{2}\left(\frac{\phi_1 + \phi_2}{\phi_1 - \phi_2}\right)$$

We see that ICR ~~also~~ will always be on the y_R axis. Hence its coordinates will be $(0, R)$.

b) For wheel 1, $\alpha_1 = -\pi/2$, $\beta_1 = \pi$, $u = \dot{x}_R$, $v = \dot{y}_R$, $\omega = \dot{\theta}_R$

For wheel 2, $\alpha_2 = \pi/2$, $\beta_2 = 0$

$r_1 = r_2 = r$, $l_1 = l_2 = b/2$

The rolling constraints of both the drive wheels are:

$$\begin{bmatrix} \sin(\alpha_1+\beta_1) & -\cos(\alpha_1+\beta_1) & -l_1\cos(\beta_1) \\ \sin(\alpha_2+\beta_2) & -\cos(\alpha_2+\beta_2) & -l_2\cos(\beta_2) \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} 1 & 0 & b/2 \\ 1 & 0 & -b/2 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{x}_R + \frac{b}{2}\dot{\theta}_R \\ \dot{x}_R - \frac{b}{2}\dot{\theta}_R \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{x}_R + \frac{b}{2}\dot{\theta}_R - r\dot{\phi}_1 \\ \dot{x}_R - \frac{b}{2}\dot{\theta}_R - r\dot{\phi}_2 \end{bmatrix} = 0 \iff \begin{bmatrix} u + \frac{b}{2}\omega - r\dot{\phi}_1 \\ u - \frac{b}{2}\omega - r\dot{\phi}_2 \end{bmatrix} = 0$$

c) The sliding constraints for both the drive wheels are:

$$\begin{bmatrix} \cos(\alpha_1+\beta_1) & \sin(\alpha_1+\beta_1) & l_1\sin(\beta_1) \\ \cos(\alpha_2+\beta_2) & \sin(\alpha_2+\beta_2) & l_2\sin(\beta_2) \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = 0$$

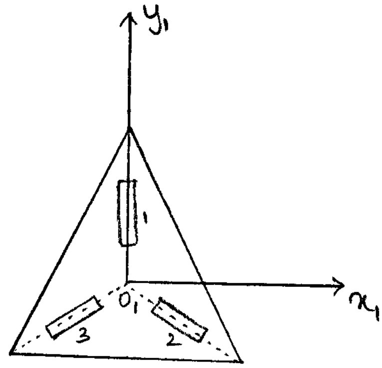
$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\theta}_R \end{bmatrix} = 0$$

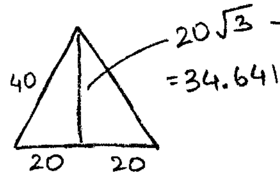
$$\begin{bmatrix} \dot{y}_R \\ \dot{y}_R \end{bmatrix} = 0 \iff \begin{bmatrix} v \\ v \end{bmatrix} = 0$$

d) The kinematic constraints are non-holonomic. A non-holonomic constraint requires a differential relation, such as the derivative of a position variable. This is consistently visible in both the kinematic constraints of the system. Hence, they are non-holonomic. This also makes the robot a non-holonomic robot.

e) For the given system, the ~~tax~~ rank of the sliding constraints matrix is 1 and the degree of mobility is 2. Hence, the robot can control both, the rate of its change of orientation, and its forward/reverse speed, simply by manipulating its wheel velocities. This implies that the ICR is constrained to lie on the infinite line extending from its wheels' horizontal axles.

Problem 3





23.094

$20/\sqrt{3} = 11.547 = l$

Here, $\alpha_1 = \dfrac{\pi}{2}$, $\beta_1 = 0$

$\alpha_2 = \dfrac{-\pi}{6}$, $\beta_2 = 0$

$\alpha_3 = \dfrac{-5\pi}{6}$, $\beta_2 = 0$.

$l_1 = l_2 = l_3 = 11.547$, $r_1 = r_2 = r_3 = r$ (assumption)

a) The rolling constraints for all wheels can be given by:

$$\begin{bmatrix} \sin(\alpha_1+\beta_1) & -\cos(\alpha_1+\beta_1) & -l_1\cos(\beta_1) \\ \sin(\alpha_2+\beta_2) & -\cos(\alpha_2+\beta_2) & -l_2\cos(\beta_2) \\ \sin(\alpha_3+\beta_3) & -\cos(\alpha_3+\beta_3) & -l_3\cos(\beta_3) \end{bmatrix} \dot{\xi}_R - \begin{bmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} 1 & 0 & -11.547\times 1 \\ -0.5 & -0.866 & -11.547\times 1 \\ -0.5 & 0.866 & -11.547\times 1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\theta}_1 \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \\ r\dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{x}_1 - 11.547\dot{\theta}_1 \\ -0.5\dot{x}_1 - 0.866\dot{y}_1 - 11.547\dot{\theta}_1 \\ -0.5\dot{x}_1 + 0.866\dot{y}_1 - 11.547\dot{\theta}_1 \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \\ r\dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{x}_1 - 11.547\dot{\theta}_1 - r\dot{\phi}_1 \\ -0.5\dot{x}_1 - 0.866\dot{y}_1 - 11.547\dot{\theta}_1 - r\dot{\phi}_2 \\ -0.5\dot{x}_1 + 0.866\dot{y}_1 - 11.547\dot{\theta}_1 - r\dot{\phi}_3 \end{bmatrix} = 0$$

b) The sliding constraints for all the wheels can be given by:

$$\begin{bmatrix} \cos(\alpha_1+\beta_1) & \sin(\alpha_1+\beta_1) & l_1 \sin(\beta_1) \\ \cos(\alpha_2+\beta_2) & \sin(\alpha_2+\beta_2) & l_2 \sin(\beta_2) \\ \cos(\alpha_3+\beta_3) & \sin(\alpha_3+\beta_3) & l_3 \sin(\beta_3) \end{bmatrix} \dot{\xi}_R = 0$$

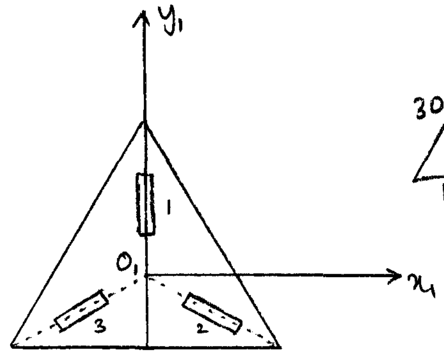
$$\begin{bmatrix} 0 & 1 & 0 \\ 0.866 & -0.5 & 0 \\ -0.866 & -0.5 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\theta}_1 \end{bmatrix} = 0$$

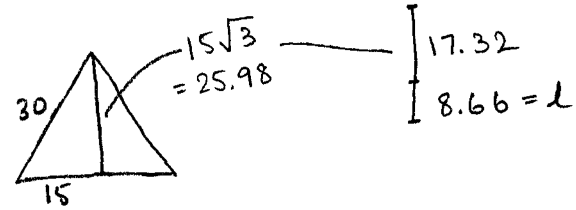
$$\begin{bmatrix} \dot{y}_1 \\ 0.866\dot{x}_1 - 0.5\dot{y}_1 \\ -0.866\dot{x}_1 - 0.5\dot{y}_1 \end{bmatrix} = 0$$

c) Here we see that the rank of the sliding constraints is 2. Hence, the degree of mobility, $\delta_m = 3-2 = 1$. This implies that the robot's motion is constrained entirely in two directions, i.e. in this case, the x_1 and y_1 directions. Therefore, the robot can control its movement ~~into~~ of only rotating ~~turning~~ about its origin, i.e.

$\dot{\theta}_1$.

Problem 4:





Here, $\alpha_1 = \frac{\pi}{2}$, $\beta_1 = \frac{\pi}{2}$

$\alpha_2 = \frac{-\pi}{6}$, $\beta_2 = \frac{\pi}{2}$

$\alpha_3 = -\frac{5\pi}{6}$, $\beta_3 = \frac{\pi}{2}$

$l_1 = l_2 = l_3 = 8.66$, $r_1 = r_2 = r_3 = r$ (assumption)

a) The rolling constraints for all wheels can be given by:

$$\begin{bmatrix} \sin(\alpha_1+\beta_1) & -\cos(\alpha_1+\beta_1) & -l_1\cos(\beta_1) \\ \sin(\alpha_2+\beta_2) & -\cos(\alpha_2+\beta_2) & -l_2\cos(\beta_2) \\ \sin(\alpha_3+\beta_3) & -\cos(\alpha_3+\beta_3) & -l_3\cos(\beta_3) \end{bmatrix} \dot{\xi}_R - \begin{bmatrix} r_1 & 0 & 0 \\ 0 & r_2 & 0 \\ 0 & 0 & r_3 \end{bmatrix} \begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} 0 & 1 & 0 \\ 0.866 & -0.5 & 0 \\ -0.866 & -0.5 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\theta}_1 \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \\ r\dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{y}_1 \cancel{-r\dot{\phi}_1} \\ 0.866\dot{x}_1 - 0.5\dot{y}_1 \\ -0.866\dot{x}_1 - 0.5\dot{y}_1 \end{bmatrix} - \begin{bmatrix} r\dot{\phi}_1 \\ r\dot{\phi}_2 \\ r\dot{\phi}_3 \end{bmatrix} = 0$$

$$\begin{bmatrix} \dot{y}_1 - r\dot{\phi}_1 \\ 0.866\dot{x}_1 - 0.5\dot{y}_1 - r\dot{\phi}_2 \\ -0.866\dot{x}_1 - 0.5\dot{y}_1 - r\dot{\phi}_3 \end{bmatrix} = 0$$

b) The sliding constraints for all the wheels can be given by:

$$\begin{bmatrix} \cos(\alpha_1+\beta_1) & \sin(\alpha_1+\beta_1) & l_1 \sin(\beta_1) \\ \cos(\alpha_2+\beta_2) & \sin(\alpha_2+\beta_2) & l_2 \sin(\beta_2) \\ \cos(\alpha_3+\beta_3) & \sin(\alpha_3+\beta_3) & l_3 \sin(\beta_3) \end{bmatrix} \dot{\xi}_R = 0$$

$$\begin{bmatrix} -1 & 0 & \cancel{l_1}\, 8.66 \times 1 \\ 0.5 & 0.866 & \cancel{l_2}\, 8.66 \times 1 \\ 0.5 & -0.866 & 8.66 \times 1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\theta}_1 \end{bmatrix} = 0$$

$$\begin{bmatrix} -\dot{x}_1 + 8.66\,\dot{\theta}_1 \\ 0.5\dot{x}_1 + 0.866\dot{y}_1 + 8.66\dot{\theta}_1 \\ 0.5\dot{x}_1 - 0.866\dot{y}_1 + 8.66\dot{\theta}_1 \end{bmatrix} = 0$$

c) Here, we see that the rank of the sliding constraints is 3. Hence, the degree of mobility, $\delta_m = 3 - 3 = 0$. This implies that the robot's motion is fully constrained in all the directions. Therefore, the robot is not capable of movement/motion at all.

Problem 5:

a)

In [231]:
```
import numpy as np
import math
```

In [232]:
```
R = lambda t: np.array([[math.cos(t), -1*math.sin(t), 0.0], [math.sin(t), math
.cos(t), 0.0], [0.0, 0.0, 1.0]])
```

In [233]:
```
r = R(np.pi/6)
r
```

Out[233]:
```
array([[ 0.8660254, -0.5      ,  0.       ],
       [ 0.5      ,  0.8660254,  0.       ],
       [ 0.       ,  0.       ,  1.       ]])
```

In [234]:
```
r.T
```

Out[234]:
```
array([[ 0.8660254,  0.5      ,  0.       ],
       [-0.5      ,  0.8660254,  0.       ],
       [ 0.       ,  0.       ,  1.       ]])
```

In [235]:
```
np.linalg.inv(r)
```

Out[235]:
```
array([[ 0.8660254,  0.5      ,  0.       ],
       [-0.5      ,  0.8660254,  0.       ],
       [ 0.       ,  0.       ,  1.       ]])
```

In [236]:
```
np.linalg.det(r)
```

Out[236]:
```
1.0
```

In [237]:
```
n = r[:,0]
s = r[:,1]
a = r[:,2]
```

In [238]:
```
np.linalg.norm(n)
```

Out[238]:
```
1.0
```

In [239]:
```
np.linalg.norm(s)
```

Out[239]:
```
1.0
```

In [240]:
```
np.linalg.norm(a)
```
Out[240]:
```
1.0
```

In [241]:
```
np.dot(n,s)
```
Out[241]:
```
0.0
```

In [242]:
```
np.dot(s,a)
```
Out[242]:
```
0.0
```

In [243]:
```
np.dot(a,n)
```
Out[243]:
```
0.0
```

b) t = pi/3

In [244]:
```
r = R(np.pi/3)
r
```
Out[244]:
```
array([[ 0.5      , -0.8660254,  0.        ],
       [ 0.8660254,  0.5      ,  0.        ],
       [ 0.       ,  0.       ,  1.        ]])
```

In [245]:
```
r.T
```
Out[245]:
```
array([[ 0.5      ,  0.8660254,  0.        ],
       [-0.8660254,  0.5      ,  0.        ],
       [ 0.       ,  0.       ,  1.        ]])
```

In [246]:
```
np.linalg.inv(r)
```
Out[246]:
```
array([[ 0.5      ,  0.8660254,  0.        ],
       [-0.8660254,  0.5      , -0.        ],
       [ 0.       ,  0.       ,  1.        ]])
```

In [247]:
```
np.linalg.det(r)
```
Out[247]:
```
1.0
```

In [248]:
```
n = r[:,0]
s = r[:,1]
a = r[:,2]
```

In [249]:
```
np.linalg.norm(n)
```
Out[249]:
```
1.0
```

In [250]:
```
np.linalg.norm(s)
```
Out[250]:
```
1.0
```

In [251]:
```
np.linalg.norm(a)
```
Out[251]:
```
1.0
```

In [252]:
```
np.dot(n,s)
```
Out[252]:
```
0.0
```

In [253]:
```
np.dot(s,a)
```
Out[253]:
```
0.0
```

In [254]:
```
np.dot(a,n)
```
Out[254]:
```
0.0
```

t = -pi/4

In [255]:
```
r = R(-1*np.pi/4)
r
```
Out[255]:
```
array([[ 0.70710678,  0.70710678,  0.        ],
       [-0.70710678,  0.70710678,  0.        ],
       [ 0.        ,  0.        ,  1.        ]])
```

In [256]:
```
r.T
```
Out[256]:
```
array([[ 0.70710678, -0.70710678,  0.        ],
       [ 0.70710678,  0.70710678,  0.        ],
       [ 0.        ,  0.        ,  1.        ]])
```

In [257]:
```
np.linalg.inv(r)
```
Out[257]:
```
array([[ 0.70710678, -0.70710678,  0.        ],
       [ 0.70710678,  0.70710678,  0.        ],
       [ 0.        ,  0.        ,  1.        ]])
```

In [258]:
```
np.linalg.det(r)
```
Out[258]:
```
1.0
```

In [259]:
```
n = r[:,0]
s = r[:,1]
a = r[:,2]
```

In [260]:
```
np.linalg.norm(n)
```
Out[260]: 1.0

In [261]:
```
np.linalg.norm(s)
```
Out[261]: 1.0

In [262]:
```
np.linalg.norm(a)
```
Out[262]: 1.0

In [263]:
```
np.dot(n,s)
```
Out[263]: 0.0

In [264]:
```
np.dot(s,a)
```
Out[264]: 0.0

In [265]:
```
np.dot(a,n)
```
Out[265]: 0.0

In [ ]:
```

```

Problem 5:

c) The following results are interpreted from the outputs:

i) $R^T = R^{-1}$

ii) $\det(R) = 1$

iii) Product of all the norm vectors are zero with each others.

These results ~~tell~~ imply that the rotation matrix is an orthogonal matrix ($R^T = R^{-1}$, $\det(R) = 1$, R is a square matrix). Also, the norm vectors are perpendicular to each other since their dot products are zero with each other.

Problem 6:

In [75]:
```
import numpy as np
```

In [76]:
```
skew = lambda v: np.array([[0.0, -v[2], v[1]], [v[2], 0.0, -v[0]], [-v[1], v[0
], 0.0]])
```

Example 1:

In [77]:
```
v_1 = [10, 12, 13]
v_2 = [2, 4, 6]
s = skew(v_1)
```

In [78]:
```
p_1 = np.cross(v_1, v_2)
p_2 = np.matmul(np.array(s), v_2)
```

In [79]:
```
print("v1 = {} and v2 = {}".format(v_1, v_2))
print("v1 x v2 = {}".format(p_1))
print("S(v1) v2 = {}".format(p_2))
print("Therefore, v1 x v2 = S(v1) v2")
```

```
v1 = [10, 12, 13] and v2 = [2, 4, 6]
v1 x v2 = [ 20 -34  16]
S(v1) v2 = [ 20. -34.  16.]
Therefore, v1 x v2 = S(v1) v2
```

Example 2:

In [80]:
```
v_1 = [24, 15, 37]
v_2 = [54, 65, 39]
s = skew(v_1)
```

In [81]:
```
p_1 = np.cross(v_1, v_2)
p_2 = np.matmul(np.array(s), v_2)
```

In [82]:
```
print("v1 = {} and v2 = {}".format(v_1, v_2))
print("v1 x v2 = {}".format(p_1))
print("S(v1) v2 = {}".format(p_2))
print("Therefore, v1 x v2 = S(v1) v2")
```

```
v1 = [24, 15, 37] and v2 = [54, 65, 39]
v1 x v2 = [-1820  1062   750]
S(v1) v2 = [-1820.  1062.   750.]
Therefore, v1 x v2 = S(v1) v2
```

Example 3:

In [83]:
```
v_1 = [0.4, 0.2, 0.7]
v_2 = [0.33, 0.5, 0.9]
s = skew(v_1)
```

In [84]:
```
p_1 = np.cross(v_1, v_2)
p_2 = np.matmul(np.array(s), v_2)
```

In [85]:
```
print("v1 = {} and v2 = {}".format(v_1, v_2))
print("v1 x v2 = {}".format(p_1))
print("S(v1) v2 = {}".format(p_2))
print("Therefore, v1 x v2 = S(v1) v2")
```

```
v1 = [0.4, 0.2, 0.7] and v2 = [0.33, 0.5, 0.9]
v1 x v2 = [-0.17  -0.129  0.134]
S(v1) v2 = [-0.17  -0.129  0.134]
Therefore, v1 x v2 = S(v1) v2
```

In [ ]:
```
```

Problem 7:

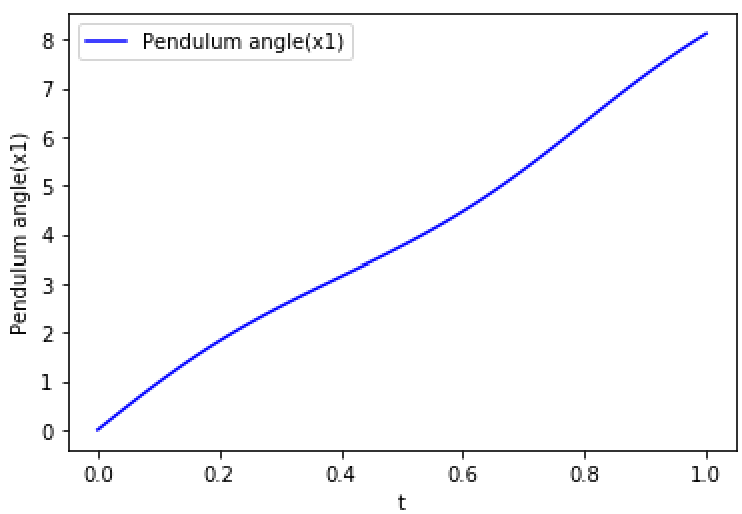
In [46]:
```
import numpy as np
import math
import matplotlib.pyplot as plt
from scipy.integrate import odeint
```

In [47]:
```
def derfunc(y, t):
    g = 32 # ft/s^2
    L = 2 # ft
    x1, x2 = y
    ydot = [0,0]
    ydot[0] = x2
    ydot[1] = -g*math.sin(x1)/L
    return ydot
```

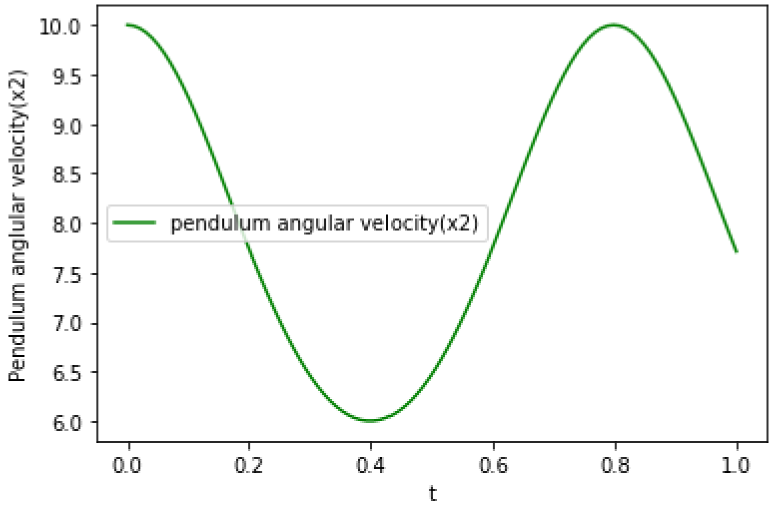
In [48]:
```
y0 = [0, 10]
t = np.linspace(0, 1, 100)
```

In [49]:
```
sol = odeint(derfunc, y0, t)
```

In [50]:
```
plt.plot(t, sol[:, 0], 'b', label='Pendulum angle(x1)')
plt.legend(loc='best')
plt.xlabel('t')
plt.ylabel('Pendulum angle(x1)')
plt.show()
```

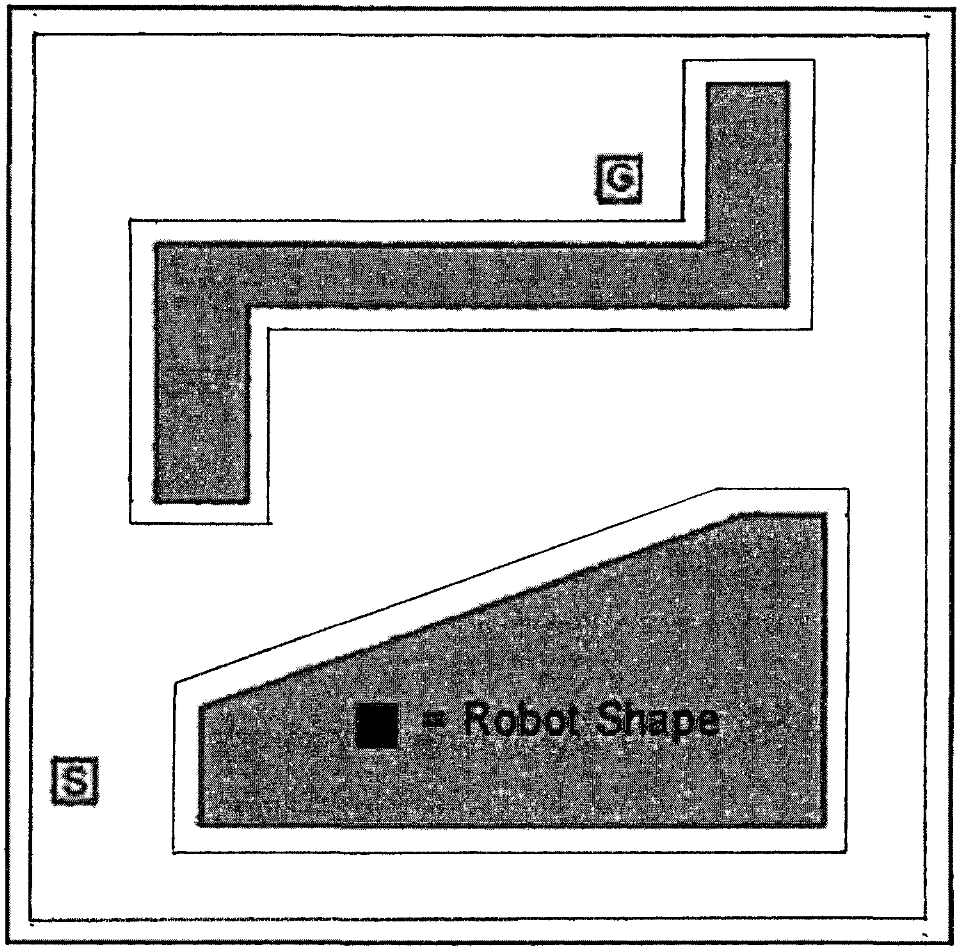


In [51]:
```
plt.plot(t, sol[:, 1], 'g', label='pendulum angular velocity(x2)')
plt.legend(loc='best')
plt.xlabel('t')
plt.ylabel('Pendulum anglular velocity(x2)')
plt.show()
```

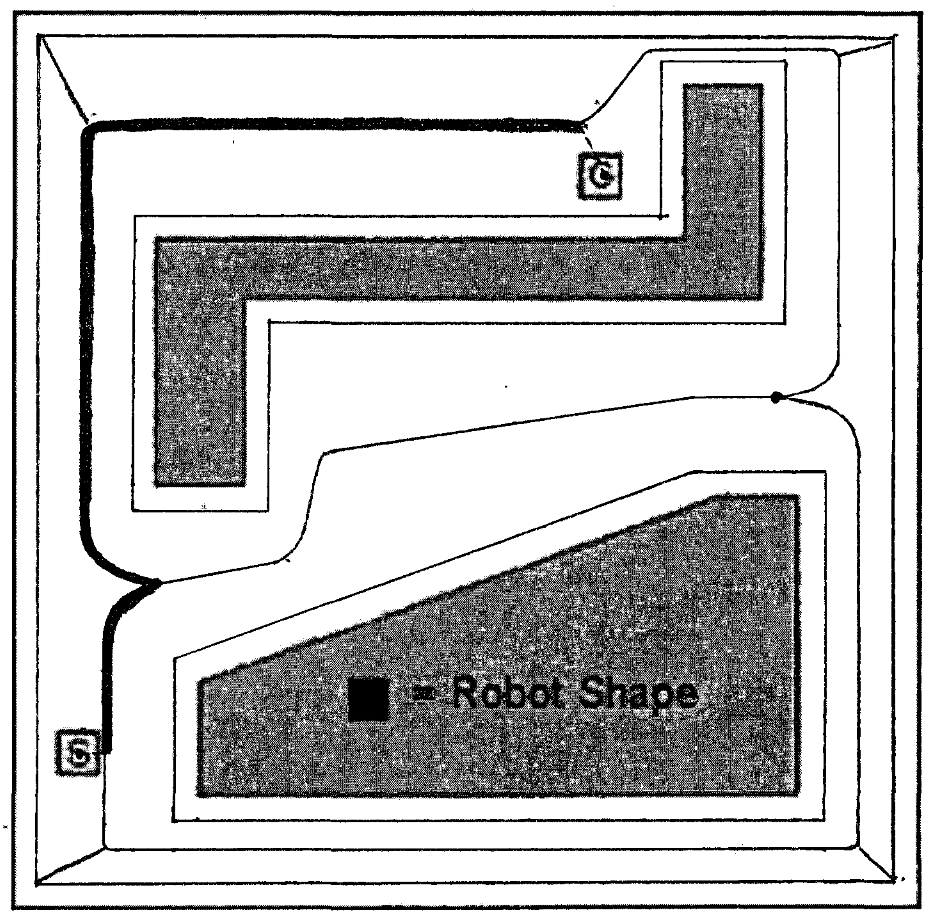


In [ ]:

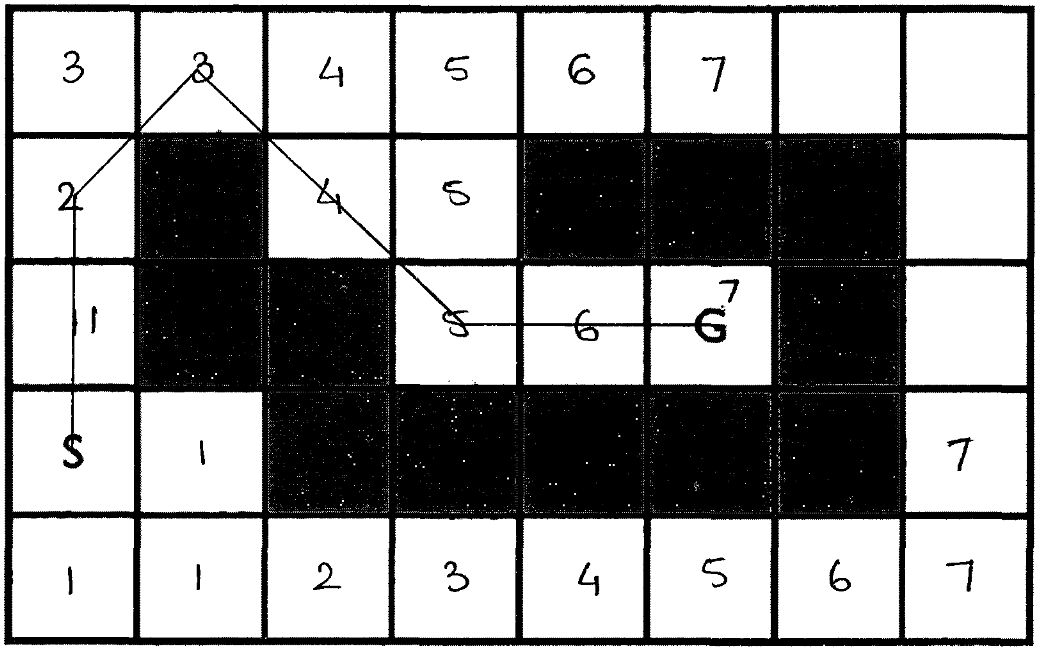
Problem 8:



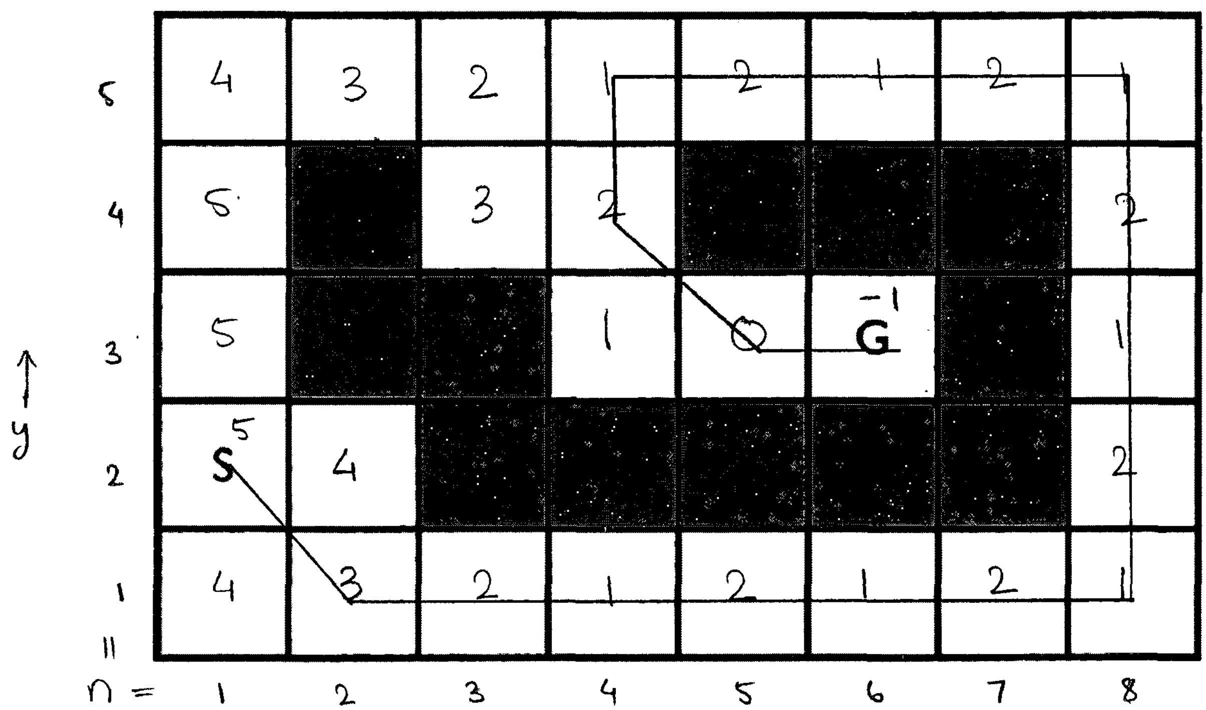
Problem 9:



Problem 10:



Problem 11:



The heuristic fn. is: $x \longrightarrow$

$$h(x) = |x_n - x_g| + |y_n - y_g| - (2 \times (\min(|x_n - x_g|, |y_n - y_g|)) - 1)$$

where x_n ~~is~~ ~~that~~, y_n are the coordinates of the point under analysis

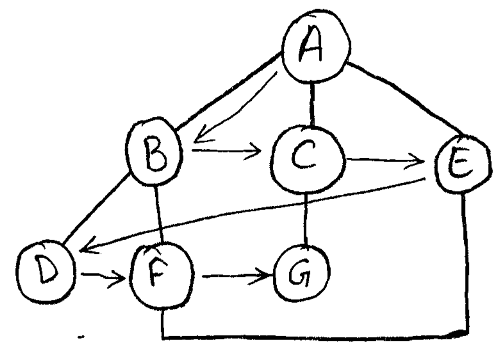
x_g, y_g are the coordinates of the ~~cell~~ goal cell.

~~max~~ min( ) is the minimum function.

Problem 12

a) The Breadth First Search applies the FIFO (First In First Out) rule in the following algorithm:

Explored	{A}	{A}	{A, B}	{A, B, C}	{A, B, C, E}	{A, B, C, E, D}	{A, B, C, E, D, F}	{A, B, C, E, D, F, G}
Frontiers	{ }	{B, C, E}	{C, E, D, F}	{E, D, F, G}	{D, F, G}	{F, G}	{G}	{ }
Unexplored	{B, C, E, D, F, G}	{D, F, G}	{F, G}	{ }	{ }	{ }	{ }	{ }

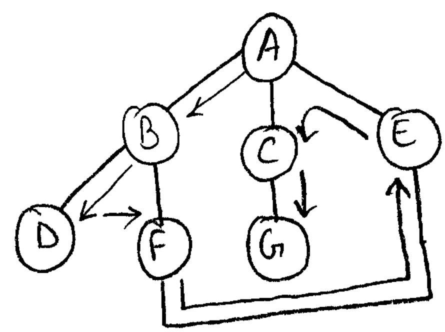


Therefore, the order in which the nodes are expanded is:

A ⟶ B ⟶ C ⟶ E ⟶ D ⟶ F ⟶ G

b) The Depth First Search applies the LIFO (Last In First Out) rule in the following algorithm:

Explored	{A}	{A}	{A, B}	{A, B, D}	{A, B, D, F}	{A, B, D, F, E}	{A, B, D, F, E, C}	{A, B, D, F, E, C, G}
Frontiers	{ }	{B, C, E}	{C, E, D, F}	{F, C, E}	{C, E}	{C}	{G}	{ }
Unexplored	{B, C, E, D, F, G}	{D, F, G}	{G}	{G}	{G}	{G}	{ }	{ }



Therefore, the order in which the nodes are expanded is:

A ⟶ B ⟶ D ⟶ F ⟶ E ⟶ C ⟶ G