5.8.30

Kartik Lahoti - EE25BTECH11032

September 30, 2025

Question

Rambha travels 300 km to her home partly by train and partly by bus. She takes 4 hours if she travels 60 km by train and the remaining by bus. If she travels 100 km by train and the remaining by bus, she takes 10 minutes longer. Find the speed of the train and the bus seperately.

Given

Symbol	Value	Description
$\mathbf{n_1}$	$\begin{pmatrix} 60 \\ 240 \end{pmatrix}$	Normal Vector
$\mathbf{n_2}$	$\begin{pmatrix} 100 \\ 200 \end{pmatrix}$	Normal Vector
c_1	4	Constant 1
c_2	$\frac{25}{6}$	Constant 2
$\mathbf{P}$	?	Reciprocal Speed Vector

Table: 5.8.30

Theoretical Solution

Let the equations be,

$$\mathbf{n_1}^\top \mathbf{X} = c_1 \tag{1}$$
$$\mathbf{n_2}^\top \mathbf{X} = c_2 \tag{2}$$

Since $\mathbf{P}$ satisfies both the lines,

$$\mathbf{n_1}^\top \mathbf{P} = c_1 \tag{3}$$
$$\mathbf{n_2}^\top \mathbf{P} = c_2 \tag{4}$$

Theoretical Solution

Solving for **P**

$$\left(\begin{array}{cc|c} 60 & 240 & 4 \\ 100 & 200 & \frac{25}{6} \end{array}\right) \xleftrightarrow{R_1 \to \frac{R_1}{60}} \left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 100 & 200 & \frac{25}{6} \end{array}\right) \tag{5}$$

$$\left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 100 & 200 & \frac{25}{6} \end{array}\right) \xleftrightarrow{R_2 \to \frac{R_2}{100}} \left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 1 & 2 & \frac{1}{24} \end{array}\right) \tag{6}$$

Theoretical Solution

$$\left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 1 & 2 & \frac{1}{24} \end{array}\right) \xleftrightarrow{R_2 \to R_2 - R_1} \left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 0 & -2 & \frac{-1}{40} \end{array}\right) \tag{7}$$

$$\left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 0 & -2 & \frac{-1}{40} \end{array}\right) \xleftrightarrow{R_2 \to \frac{R_2}{-2}} \left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 0 & 1 & \frac{1}{80} \end{array}\right) \tag{8}$$

Theoretical Solution

$$\left(\begin{array}{cc|c} 1 & 4 & \frac{1}{15} \\ 0 & 1 & \frac{1}{80} \end{array}\right) \xleftrightarrow{R_1 \to R_1 - 4R_2} \left(\begin{array}{cc|c} 1 & 0 & \frac{1}{60} \\ 0 & 1 & \frac{1}{80} \end{array}\right) \tag{9}$$

$$\therefore \mathbf{P} = \begin{pmatrix} \frac{1}{60} \\ \frac{1}{80} \end{pmatrix} \tag{10}$$

Theoretical Solution

Since **P** is the reciprocal of the speeds
The speed of train is $60\,km/h$ and bus is $80\,km/h$

C Code - To find inverse of a Matrix

```c
#include <stdio.h>
#include <math.h>
void row_mal(double X[][3] , double k , int n , int m)
{
    for(int i = 0 ; i< 3 ; i++)
    {
        X[n][i] = X[n][i] - k * X[m][i];
    }
}

void row_div(double X[][3] , double k , int n )
{
    for(int i = 0 ; i< 3 ; i++)
    {
        X[n][i] /= k ;
    }
}
```

C Code

```
void augment(double *A , double *B , double *C)
{
    double X[2][3];
    for(int i = 0; i< 2 ; i++){
        X[i][0] = A[i];
        X[i][1] = B[i];
        X[i][2] = C[i];
    }
    if(X[0][0] != 0 ){
        row_div(X,X[0][0],0);
        if(X[1][0] != 0)
            row_mal(X,X[1][0],1,0);
    }
    else{
        row_mal(X,-1,0,1);
        row_div(X,X[0][0],0);
        if(X[1][0] != 0)
            row_mal(X,X[1][0],1,0);
    }
```

C Code

```
    if(X[1][1] != 0 ){
        row_div(X,X[1][1],1);
        if(X[0][1] != 0)
            row_mal(X,X[0][1],0,1);
    }
    else{
        row_mal(X,-1,1,0);
        row_div(X,X[1][1],1);
        if(X[0][1] != 0)
            row_mal(X,X[0][1],0,1);
    }
    for(int i = 0 ; i< 2 ; i++){
        C[i] = 1 / X[i][2] ;
        for(int j = 0; j < 3; j++)
            printf("%.3f ",X[i][j]);
        printf("\n");
    }
}
```

C Code - To generate Line

```
void linegen(double *XY, double *A , double *B , int n , int m )
{
    double temp[m] ;
    for (int i = 0 ; i < m ; i++)
    {
        temp [ i ] = (B[i]- A[i]) /(double) n ;
    }
    for (int i = 0 ; i < n ; i++ )
        for (int j = 0 ; j < m ; j++)
            XY[j*n + i ] = A[j] + temp[j] * i ;

}
```

Python Code

```
import ctypes as ct
import numpy as np
import matplotlib.pyplot as plt

handc1 = ct.CDLL("./func.so")

handc1.augment.argtypes = [
    ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double),
    ct.POINTER(ct.c_double)
]
handc1.augment.restype = None
A = np.array([60,100] , dtype = np.float64).reshape(-1,1)
B = np.array([240,200] , dtype = np.float64).reshape(-1,1)
C = np.array([4,25/6], dtype = np.float64).reshape(-1,1)
```

Python Code

```
handc1.augment(
    A.ctypes.data_as(ct.POINTER(ct.c_double)),
    B.ctypes.data_as(ct.POINTER(ct.c_double)),
    C.ctypes.data_as(ct.POINTER(ct.c_double))
)

print("Speed of Train : ",C[0]);
print("Speed of Bus : ",C[1]);
```

Python Code

```
def line(P: np.ndarray , Q: np.ndarray, str1 , str2):
    handc2 = ct.CDLL("./line_gen.so")

    handc2.linegen.argtypes = [
        ct.POINTER(ct.c_double),
        ct.POINTER(ct.c_double),
        ct.POINTER(ct.c_double),
        ct.c_int , ct.c_int
    ]

    handc2.linegen.restype = None
```

Python Code

```
n = 200
    XY = np.zeros((2,n),dtype=np.float64)

    handc2.linegen (
        XY.ctypes.data_as(ct.POINTER(ct.c_double)),
        P.ctypes.data_as(ct.POINTER(ct.c_double)),
        Q.ctypes.data_as(ct.POINTER(ct.c_double)),
        n,2
    )
    plt.plot(XY[0,:],XY[1,:], str1 , label = str2 )
```

Python Code

```
plt.figure()
M = np.array([61/15,-1],dtype=np.float64).reshape(-1,1)
N = np.array([-10,151/60],dtype=np.float64).reshape(-1,1)
line(M,N,"g-","Line 1 ")
M = np.array([2+1/24 , -1],dtype=np.float64).reshape(-1,1)
N = np.array([-10,5+1/48],dtype=np.float64).reshape(-1,1)
line(M,N,"r-","Line 2")
plt.scatter(1/np.squeeze(C[0]),1/np.squeeze(C[1]))
plt.annotate(f"P\n(1/{np.squeeze(C[0]):.0f},1/{np.squeeze(C[1])
    :.0f})",(1/np.squeeze(C[0]),1/np.squeeze(C[1])),textcoords =
    "offset points" ,xytext = (0,-25),ha = "center")
```

Python Code

```
plt.xlim([-1/2,1/2])
plt.ylim([-1/2,1/2])
plt.xlabel("$x$")
plt.ylabel("$y$")
plt.grid()
plt.legend(loc="best")
plt.title("5.8.30")

plt.savefig("../figs/Inter1.png")
plt.show()
#plt.savefig('../figs/Inter1.png')
#subprocess.run(shlex.split("termux-open ../figs/Inter1.png"))
```

Python Code - 2

```
import math
import sys
sys.path.insert(0, '/home/kartik-lahoti/matgeo/codes/CoordGeo')
import numpy as np
import numpy.linalg as LA
import matplotlib.pyplot as plt

from line.funcs import *
A = np.array([[60,240],[100,200]] , dtype = np.float64)
C = np.array([4,25/6], dtype = np.float64).reshape(-1,1)
sol = LA.solve(A,C)

print("Speed of Train = " , 1/sol[0])
print("Speed of Bus = " , 1/sol[1])
```

Python Code - 2

```
def plot_it(P,Q,str1,str2):
    x_l = line_gen_num(P,Q,20)
    plt.plot(x_l[0,:],x_l[1,:] , str1 , label = str2)

plt.figure()
M = np.array([61/15,-1],dtype=np.float64).reshape(-1,1)
N = np.array([-10,151/60],dtype=np.float64).reshape(-1,1)
plot_it(M,N,"g-","Line 1 ")
M = np.array([2+1/24 , -1],dtype=np.float64).reshape(-1,1)
N = np.array([-10,5+1/48],dtype=np.float64).reshape(-1,1)
plot_it(M,N,"r-","Line 2")
```

Python Code - 2

```
plt.scatter(np.squeeze(sol[0]),np.squeeze(sol[1]))
plt.annotate(f"P\n(1/{1/np.squeeze(sol[0]):.0f},1/{1/np.squeeze(
    sol[1]):.0f})",(np.squeeze(sol[0]),np.squeeze(sol[1])),
    textcoords = "offset points" ,xytext = (0,-25),ha = "center")

plt.xlim([-1/2,1/2])
plt.ylim([-1/2,1/2])

plt.xlabel("$x$")
plt.ylabel("$y$")
plt.grid()
```

Python Code - 2

```
plt.legend(loc="best")

plt.title("5.8.30")

plt.savefig("../figs/Inter2.png")
plt.show()

#plt.savefig('../figs/Inter2.png')
#subprocess.run(shlex.split("termux-open ../figs/Inter2.png"))
```

Plot