**Variables and Operators – CC**

Spherical Cargos

side=input()

d=input()

n=input()

Vol=side\*side\*side

r=float(d)/2

Vol1=(4\*3.14\*r\*r\*r)/3

tot\_vol=Vol1\*n

rem=Vol-tot\_vol

print "%.2f" % rem

Celsius to Fahrenheit Converter

c=input("Enter the Celsius :")

f=(c \*1.8 ) + 32

print "%(c).1f degree Celsius is equal to %(f).1f degree Fahrenheit" % {"c":c ,"f":f}

**Decision Making – CC**

Offer to Valuable Customers

r=raw\_input("Enter the type of relationship with customer\n")

s=raw\_input("Enter the slot of the customer\n")

if (r=="New"):

dis=float(10)/100

elif (r=="Gold"):

dis=float(25)/100

if (r=="Platinum"):

dis=float(40)/100

if (s=="S1") or (s=="S3") :

f=200

elif (s=="S2") or (s=="S4"):

f=800

else:

f="Invalid Input"

if f!="Invalid Input":

final\_fare = f - ( f \* dis)

print("Amount to be paid = %d" % final\_fare )

else:

print"Invalid Input"

Discount Tuesdays

t1=input()

t2=input()

b1=input()

b2=input()

f1=input()

f2=input()

c1=input()

c2=input()

wt =(t1\*c1) + ( t2 \*c2)

wb=(b1\*c1) + ( b2 \*c2)

wf=(f1\*c1) + ( f2 \*c2)

if ( wt < wb ) and ( wt < wf):

print "Train Transportation"

if ( wb < wf ) and ( wb < wt):

print "Bus Transportation"

if ( wf < wt ) and ( wf < wb):

print "Flight Transportation"

**Creating and Executing Loops –CC**

Cargo Id and Port Id

val=input()

result=list()

for x in range(1,(val+1)):

if val%x == 0:

result.append(x)

result.sort()

print " ".join(str(z) for z in result)

The Container Contains

j=0

while True:

print("Enter the number of items in the box "+str(j+1))

count=input()

j+=1

if count%8 !=0:

break

print("Number of boxes stored in container is "+str(j-1))

**Working with Lists-CC**

All Products

data1=raw\_input("Enter the Numbers in List-1\n")

list1=data1.split(" ")

data2=raw\_input("Enter the Numbers in List-2\n")

list2=data2.split(" ")

while '' in list1:

list1.remove('')

while '' in list2:

list2.remove('')

result=list()

for x in range(len(list1)):

num=int(list1[x])

for y in range(len(list2)):

mu=num\*int(list2[y])

if mu%2 !=0:

result.append(mu)

if len(result)>0:

print " ".join(str(z) for z in result)

else:

print "No such Elements in the list"

Rotate the List

import collections

data=raw\_input("Enter the Numbers in List\n")

myList=data.split(" ")

print "Before Rotating : "

print " ".join(str(y) for y in myList)+" "

d=collections.deque(myList)

d.rotate(-1)

print "After Rotating : "

print " ".join(str(y) for y in d)

**Working with Strings-CC**

text=raw\_input()

if text.find("medicine")>=0 or text.find("tablet")>=0 or text.find("drugs")>=0:

print "C-Cargo"

elif text.find("chocolate")>=0 or text.find("meat")>=0 or text.find("fruit")>=0:

print "F-Cargo"

elif text.find("electronics")>=0 or text.find("mobile")>=0 or text.find("PC")>=0:

print "E-Cargo"

**Working with Tuples-CC**

data=raw\_input()

nu=data.split(",")

count=0

myList=[]

while count<len(nu):

newInt=int(nu[count])

count=count+1

if newInt not in myList:

myList.append(newInt)

print tuple(myList)

**Declaring Functions and Passing Values-CC**

f=raw\_input()

s=raw\_input()

fl=f.strip().split(',')

sl=s.strip().split(',')

print fl

print sl

fll=[x.lower() for x in fl]

sll=[x.lower() for x in sl]

if any(i in fll for i in sll):

print "Overlapping"

else:

print "Non Overlapping"

**Working with Dictionary-CC**

in1=raw\_input()

in2=raw\_input()

myList=in1.split(",")

dict={}

for x in range(len(myList)):

sp=myList[x].split()

key=sp[0]

value=sp[1]

dict[key]=value

def convert(param):

val=''

p=param.split()

for y in range(len(p)):

if p[y] not in dict.keys():

print "The sentence cannot be translated"

exit()

else:

val=val+' '+dict[p[y]]

if val!='':

print val

convert(in2)

**File Handling-CC**

[Smallest Cargo Name](https://qa2qe.cognizant.e-box.co.in/problem/showDescription/4478)

f=open('text.txt','r')

myLine=f.read().splitlines()

myLine.sort(key=len)

min=len(myLine[0])

s=list()

for i in range (len(myLine)):

if len(myLine[i])==min:

s.append(myLine[i])

for j in range(len(s)):

print s[j]

[Eligible Candidate](https://qa2qe.cognizant.e-box.co.in/problem/showDescription/4496)

import xml.etree.ElementTree as ET

tree=ET.parse('candidate.xml')

root=tree.getroot()

for can in root.findall('candidate'):

if int(can.find('age').text)>=25:

print can.find('candidateName').text+" : "+can.find('age').text