Pseudo Code

204234F Wijesekera C.K

Module main()

input,rotate[2],rotate[0]='y'

While if rotate[0]='y'

Do you want to know?

1) Price elasticity of demand and Total Revenue (Enter 1)
2) Price elasticity of supply (Enter 2)
3) Income elasticity of demand (Enter 3)
4) Cross elasticity of demand (Enter 4)
5) Available budget to maximize (Enter 5)

Then Enter your choice : input

if (input == 1)

call price_demand()

else if (input == 2)

call price_supply()

else if (input == 3)

call income_demand()

else if (input == 4)

call cross_demand()

else if (input == 5)

call maximize_budget()

you want to end this program?(yes/no)

End While

End main modul

Module Price_demand()

BEGIN

Enter your price and changed price : f_price c_price

Enter your quantity demand and changed quantity demand : f_demand c_demand

Process

PED = ((c_demand-f_demand) / (c_price-f_price))*(f_price/f_demand)

Output

Price Elasticity Of Demand value

If(PED>1)

This is an Elastic.

If (c_price>f_price)

Total revenue is Decrease.

else

Total revenue is Increase.

End If

If(PED<1 && PED >0)

This is an Inelastic

if(c_price>f_price)

Total revenue is Increase.

else

Total revenue is Decrease.

End If

You want to know idea about revenue values? (yes/no) : idea

If idea = “yes”

Process

R1= f_price*f_demand

R2= c_price*c_demand

Total_revenue=R2-R1

Output

First revenue = R1

Second revenue = R2

Increase revenue= R2-R1

If idea = “no”

No process, end this function.

END

End module Price_demand

Module Price_supply()

BEGIN

Enter your price and changed price : f_price c_price

Enter your quantity supply and changed quantity supply : f_supply c_supply

Process

PES = ((c_supply-f_supply) / (c_price-f_price))*(f_price/f_supply)

Output

Price Elasticity Of Supply Value.

If(PES>1)

This is an Elastic.

Supply change value is large.

End If

If(PES<1)

This is an Inelastic.

Supply change value is small.

End If

End

End module Price_supply()

Module income_demand()

BEGIN

Enter your income and changed income : f_income c_income

Enter your quantity demand and changed quantity demand : f_demand c_demand

Process

Q=(c_demand-f_demand)/f_demand;

I=(c_income-f_income)/f_income;

YED=Q/I

Output

Income elasticity of demand Value

IF(YED>0) **THEN**

This good is Normal good.

Else

This good is Inferior good.

ENDIF

END

End module income_demand

Module cross_demand()

BEGIN

Enter your food X quantity demand and changed demand : x1_demand x2_demand

Enter your food Y price and changed price : y1_price y2_price

Process

Q=(x2_demand-x1_demand)/x1_demand

P=(y2_price-y1_price)/y1_price

XED=Q/P

Output

Cross elasticity of demand Value

IF(XED>0) **THEN**

X and Y good are substitute goods.

ELSE IF(XED<0) **THEN**

X and Y good are complementary goods.

ELSE

No relationship between X and Y goods.

ENDIF

END

End module cross_demand

Module maximize_buget()

BEGIN

Enter available budget : budget

Enter Price of good X and Y : price_x price_y

Enter Unit consumed : unit

You know total utility or marginal utility (t/m):

If you enter 'm'

Enter marginal utility

- -For loop--

For i 0 to 4

x[i] , y[i] get

End For loop

Process

--For loop—

For i 0 to (unit-1) do

xu=x[i]/price_x

For j 0 to (unit-1) do

yu=y[j]/price_y

IF(xu==yu) **THEN**

IF(((i+1.0)*price_x) + ((j+1.0)*price_y) == budget) **THEN**

Consumption bundle is X value Y value

goto jump1

ENDIF

End For loop

jump1

Output

Consumption bundle is X= number of X good Y= number of X good

If you enter ‘t’

Enter Total utility

- -For loop--

For i 0 to 4

x[i] , y[i] get

End For loop

Process

--For loop—

For k 0 to (unit-1) do

xx[k] = x[k] - x[k - 1]

yy[k] = y[k] - y[k - 1]

End For loop

--For loop--

For i 0 to (unit-1) do

xu=xx[i]/price_x

For j 0 to (unit-1) do

yu=yy[j]/price_y

IF(xu==yu) **THEN**

IF(((i+1.0)*price_x) + ((j+1.0)*price_y) == budget) **THEN**

Consumption bundle is X= number of X good Y= number of X good

goto jump2

ENDIF

End For loop

Jump2

Output

Consumption bundle is X= number of X good Y= number of X good

END

END module maximize_buget