SOLUTION PROBLEM F

```
set    PROC          ordered                                                  ; # Production, Assembly, Packaging
set    TYPE          ordered                                                  ; # Electric or Gas
set    SOUR          ordered                                                  ; # Inhouse, Another Source

param  Hrq           {i in PROC, j in TYPE}                                   ; # Hours required per process per type
param  Cst           {k in SOUR, j in TYPE}                                   ; # Cost per Source per type
param  Cap           {i in PROC}                                              ; # Capacity per Process
param  Dmd           {j in TYPE}                                              ; # Demand per Type

var    QT            {k in SOUR, j in TYPE} integer >= 0                      ; # Quantity per Type


minimize TotCst      : sum {k in SOUR, j in TYPE} QT[k,j] * Cst[k,j]          ; # Minimize Total Cost

s.t. CapCon          {i in PROC} : sum {j in TYPE} QT['Inho',j] * Hrq[i,j] <= Cap[i]  ; # QT limited by Capacity per Process
s.t. DmdCon          {j in TYPE} : sum {k in SOUR} QT[k,j] >= Dmd[j]          ; # QT must meet the Demand


data                                                                          ;

set    PROC          :=          Prod              Assm              Pack     ;
set    TYPE          :=          Elec              Gas                        ;
set    SOUR          :=          Inho              Ansc                       ;

param  Hrq           :           Elec              Gas                        :=
                       Prod      0.20              0.40
                       Assm      0.30              0.50
                       Pack      0.10              0.10                       ;

param  Cst           :           Elec              Gas                        :=
                       Inho      55                85
                       Ansc      67                95                         ;

param  Cap           :=          Prod   10000  Assm   15000  Pack   5000      ;
param  Dmd           :=          Elec   30000  Gas    15000                    ;


option solver gurobi                          ;
solve                                         ;
option display_width 200, display_1col 0      ;

display QT                                    ;
```

SOLUTION PROBLEM G simple

```
set    MACH          ordered                                          ; # Machines
set    TABL          ordered                                          ; # Table Types

param  Prt           {i in MACH, j in TABL}                           ; # Production Time per Table per Machine
param  Cap           {i in MACH}                                      ; # Production Time Available per Machine per week
param  Spr           {j in TABL}                                      ; # Selling Price per Table type
param  Req           {j in TABL}                                      ; # Minimum Requirement per Table type

var    QP            {j in TABL} integer >= 0                         ; # Quantity Produced per Table type per week


maximize TotRev     : sum {j in TABL} QP[j] * Spr[j]                                ; # Maximize Total Revenue

s.t. CapCon         {i in MACH} : sum {j in TABL} QP[j] * Prt[i,j] <= Cap[i]         ; # Machine Capacity constraint
s.t. ReqCon         {j in TABL} : QP[j] >= (QP['Country'] + QP['Contemp']) * Req[j]  ; # Percentage per Table Type constraint


data                                                                 ;

set    MACH          :=     Router             Sander       Polish    ;
set    TABL          :=     Country            Contemp                ;

param  Prt           :      Country            Contemp                :=
                     Router     1.5                2.0
                     Sander     3.0                4.5
                     Polish     2.5                1.5                ;

param  Cap           :=     Router       1000  Sander  2000  Polish 1500  ;
param  Spr           :=     Country       350  Contemp       450         ;
param  Req           :=     Country       0.2  Contemp       0.3         ;


option solver gurobi                          ;
solve                                         ;
option display_width 200, display_1col 0      ;

display QP                                    ;
```

SOLUTION PROBLEM G con dummy indexes

```
set    MACH          ordered                                          ; # Machines
set    TABL          ordered                                          ; # Table Types

param  Prt           {i in MACH, j in TABL}                           ; # Production Time per Table per Machine
param  Cap           {i in MACH}                                      ; # Production Time Available per Machine per week
param  Spr           {j in TABL}                                      ; # Selling Price per Table type
param  Req           {j in TABL}                                      ; # Minimum Requirement per Table type

var    QP            {j in TABL} integer >= 0                         ; # Quantity Produced per Table type per week


maximize TotRev     : sum {j in TABL} QP[j] * Spr[j]                      ; # Maximize Total Revenue

s.t. CapCon         {i in MACH} : sum {j in TABL} QP[j] * Prt[i,j] <= Cap[i]   ; # Machine Capacity constraint
s.t. ReqCon         {j in TABL} : QP[j] >= (sum {l in TABL} QP[l]) * Req[j]    ; # Percentage per Table Type constraint


data                                                                 ;

set    MACH          :=     Router             Sander       Polish       ;
set    TABL          :=     Country            Contemp                   ;

param  Prt           :      Country            Contemp                   :=
                     Router     1.5                2.0
                     Sander     3.0                4.5
                     Polish     2.5                1.5                   ;

param  Cap           :=     Router       1000   Sander  2000  Polish 1500   ;
param  Spr           :=     Country       350   Contemp       450          ;
param  Req           :=     Country       0.2   Contemp       0.3          ;


option solver gurobi                             ;
solve                                            ;
option display_width 200, display_1col 0         ;

display QP                                       ;
```

SOLUTION PROBLEM H

```
set    VITA          ordered                                                  ; # Vitamins: A, B
set    INGR          ordered                                                  ; # Ingridients: Eggs, Bacon, Cereal

param  Con           {i in VITA, j in INGR}                                   ; # Vitamins Contribution per Ingridient
param  Req           {i in VITA}                                              ; # Minimum Requirement per Vitamin
param  Cst           {j in INGR}                                              ; # Cost per Ingridient

var    QI            {j in INGR} integer >= 0                                 ; # Quantity per Ingridient


minimize TotCst      : sum {j in INGR} QI[j] * Cst[j]                         ; # Minimize Total Cost

s.t. ReqCon          {i in VITA} : sum {j in INGR} QI[j] * Con[i,j] >= Req[i] ; # Minimum Requirement per Vitamin


data                                                                          ;

set    VITA          :=     VitA          VitB                               ;
set    INGR          :=     Eggs          Bacon          Cereal              ;

param  Con           :      Eggs          Bacon          Cereal       :=
                     VitA      2             4              1
                     VitB      3             2              1              ;


param  Req           :=     VitA     16   VitB     12                        ;
param  Cst           :=     Eggs   0.04   Bacon  0.03   Cereal 0.02         ;


option solver gurobi                          ;
solve                                         ;
option display_width 200, display_1col 0      ;

display QI                                    ;
```