More Linear Programming Models

##### The Assembly Problem - Primal Algebra



Objective: Maximize the return summed over all the final

products produced less the cost of the component parts purchased.

Constraints: The first constraint equation is a supply‑demand balance and constrains the usage of the component parts to be less than or equal to inventory plus purchases.

The second constraint limits the resources used in manufacturing final products and purchasing component parts to the exogenous resource endowment.

The last constraint imposes a minimum sales requirement on final product production



The dual problem is not very much different from those before, thus, suppose we only look at the dual constraint associated with Qk. That constraint



where Uk is the return to one unit of component part k; and Zi is the return to one more unit of limited resource I.

 This constraint is more easily interpreted if it is rewritten as follows



or, equivalently,



This inequality says that the internal value of a component part unit is less than or equal to its purchase price plus the cost of the resources used in its acquisition. Therefore, the internal value of a component part can be greater than the amount paid externally.

### More Linear Programming Models

###### The Assembly Problem – An Example

|  |  |
| --- | --- |
| **Table 7.1. Data for Computer Excess Example** |  |
| Components Required to Assemble a System |  |
|  | XT | AT | 386SX | 38633 | 486SX | 48633 |
| 360FLOPPY | 1 | 1 |  |  |  |  |
| 12MFLOPPY | 1 | 1 | 2 | 1 | 1 | 1 |
| 144MFLOPPY |  |  |  | 1 | 1 | 1 |
| HARDDISK |  | 1 | 1 | 1 | 1 | 1 |
| MONO | 1 | 1 | 1 |  |  |  |
| COLORVGA |  |  |  | 1 | 1 | 1 |
| PLAINCASE | 1 | 1 | 1 |  |  |  |
| FANCYCASE |  |  |  | 1 | 1 | 1 |
| **Components Parts Acquisition Information**  |  |
| Name  | Cost | Inventory | Labor | Shelf Space |  |  |
| 360KFLOPPY | 35 | 20 | 0.01 | 0.01 |  |  |
| 12MFLOPPY | 49 | 29 | 0.01 | 0.01 |  |  |
| 144MFLOPPY | 52 | 32 | 0.01 | 0.01 |  |  |
| HARDDISK | 245 | 45 | 0.03 | 0.03 |  |  |
| MONO | 102 | 15 | 0.07 | 1.50 |  |  |
| COLORVGA | 302 | 45 | 0.10 | 2.00 |  |  |
| PLAINCASE | 41 | 11 | 0.15 | 1.70 |  |  |
| FANCYCASE | 80 | 12 | 0.12 | 1.70 |  |  |
| **Final Products Assembly and Sales Information** |  |
| Name | Sales Price | Minimum Sales | Assembly Cost | Labor | Space |  |
| XT | 689 | 1 | 59 | 2.00 | 1 |  |
| AT | 992 | 3 | 102 | 2.05 | 1 |  |
| 386SX | 1200 | 2 | 100 | 2.21 | 1 |  |
| 38633 | 1400 | 4 | 300 | 2.24 | 1 |  |
| 486SX | 1500 | 2 | 400 | 2.18 | 1 |  |
| 48633 | 1800 | 2 | 700 | 2.12 | 1 |  |

### More Linear Programming Models

###### The Assembly Problem – An Example

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 7.2. Tableau of Computer Excess Example** |  |  |  |
|  |  |  | Assembly |  |  |  |  |  | Buy |  |  |  |  |
|  | XT | AT | 386SX | 38633 | 486SX | 48633 | 360k | 12M | 144M | HARD | MONO | CVGA | PLAIN | FANCY | RHS |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| OBJECTIVE | 630 | 890 | 1100 | 1100 | 1100 | 1100 | -35 | -49 | -52 | -245 | -102 | -302 | -41 | -80 | Max |
| 360KFLOPPY | 1 | 1 |  |  |  |  | -1 |  |  |  |  |  |  |  | ≤ | 20 |
| 12MFLOPPY | 1 | 1 | 2 | 1 | 1 | 1 |  | -1 |  |  |  |  |  |  | ≤ | 29 |
| 144MFLOPPY |  |  |  | 1 | 1 | 1 |  |  | -1 |  |  |  |  |  | ≤ | 32 |
| HARDDISK |  | 1 | 1 | 1 | 1 | 1 |  |  |  | -1 |  |  |  |  | ≤ | 45 |
| MONO | 1 | 1 | 1 |  |  |  |  |  |  |  | -1 |  |  |  | ≤ | 15 |
| COLORVGA |  |  |  | 1 | 1 | 1 |  |  |  |  |  | -1 |  |  | ≤ | 45 |
| PLAINCASE | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  | -1 |  | ≤ | 11 |
| FANCYCASE |  |  |  | 1 | 1 | 1 |  |  |  |  |  |  |  | -1 | ≤ | 12 |
| LABOR | 2 | 2.05 | 2.21 | 2.24 | 2.18 | 2.12 | 0.01 | 0.01 | 0.01 | 0.03 | 0.07 | 0.1 | 0.15 | 0.12 | ≤ | 550 |
| SHELFSPACE |  |  |  |  |  |  | 0.01 | 0.01 | 0.01 | 0.03  |  1.5 | 2.0 |  1.7 |  1.7 | ≤ | 590 |
| SYSTEMSPC | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  | ≤ | 240 |
| Lower Bound | 1 | 3 | 2 | 4 | 2 | 2 |  |  |  |  |  |  |  |  |  |  |

### More Linear Programming Models

###### The Assembly Problem – An Example

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 7.3. Solution for Computer Excess Example**  |  |  |  |
| Objective 155330.097  |
| Variable | Value | Reduced Cost |  | Constraint | Slack | Shadow Price | Level |
| XT | 1.0 | -168.4 |  | 360KFLOPPY | 16 | 0 |  |
| AT | 3.0 | -159.1 |  | 12MFLOPPY | 0 | 50.9 |  |
| 386SX | 172.9 | 0.0 |  | 144MFLOPPY | 0 | 53.9 |  |
| 38633 | 41.0 | 0.0 |  | HARDDISK | 0 | 250.7 |  |
| 486SX | 2.0 | 0.0 |  | MONO | 0 | 385.4 |  |
| 48633 | 2.0 | 0.0 |  | COLORVGA | 0 | 343.4 |  |
| 360KFLOPPY | 0.000 | -36.9 |  | PLAINCASE | 0 | 362.2 |  |
| 12MFLOPPY | 365.772 | 0.0 |  | FANCYCASE | 0 | 401.2 |  |
| 144MFLOPPY | 13.000 | 0.0 |  | LABOR | 10.09 | 0 |  |
| HARDDISK | 175.886 | 0.0 |  | SHELFSPACE | 0 | 188.9 |  |
| MONO | 161.886 | 0.0 |  | SYSTEMSPC | 18.11 | 0 |  |
| COLORVGA | 0.000 | -336.5 |  |  |  |  |  |
| PLAINCASE | 165.886 | 0.0 |  |  |  |  |  |
| FANCYCASE | 33.0 | 0.0 |  |  |   |  |  |

### More Linear Programming Models

###### The Disassembly Problem – Primal Algebra



Objective: The objective function maximizes operating

profit, which is the sum over all final products

sold (QK) of the total revenue earned by sales

less the costs of all purchased inputs.

Constraints: The first constraint is a product balance ­limiting the quantity sold to be no greater than the quantity supplied when the raw product is disassembled.

The next constraint is a resource limitation constraint on raw product disassembly and product sale.

This is followed by an upper bound on disassembly as well as upper and lower bounds on sales.

### More Linear Programming Models

###### The Disassembly Problem – An Example

|  |
| --- |
| **Table 7.4. Data for Jeremiah Junk Yard Example** |
| **Car Data** | ESCORTS | 626S | TBIRDS | CADDIES |  |
| PURCHASE PRICE | 85 | 90 | 115 | 140 |  |
| WEIGHT | 2300 | 2200 | 3200 | 3900 |  |
| DISASSEMBLY COST | 100 | 120 | 150 | 170 |  |
| AVAILABILITY | 13 | 12 | 20 | 10 |  |
|  |  |  |  |  |  |
| **Resource Use to Breakdown Cars** |
| CAPACITY | 1 | 1 | 1.2 | 1.4 |  |
| LABOR | 10 | 12 | 15 | 20 |  |
|  |  |  |  |  |  |
| **Proportional Breakdown of Cars into Component Parts** |
|  | ESCORTS | 626S | TBIRDS | CADDIES |  |
| METAL | .60 | .55 | .60 | .62 |  |
| SEATS | .10 | .10 | .06 | .04 |  |
| CHROME | .05 | .05 | .09 | .14 |  |
| DOORS | .08 | .10 | .10 | .07 |  |
| JUNK | .17 | .20 | .15 | .13 |  |
|  |  |  |  |  |  |
| **Part Data** | MINIMUM | MAXIMUM | PRICE | PARTSPACE | LABOR |
| METAL | 0 |  | 0.15 | 0 | 0.0010 |
| SEATS | 4000 | 6000 | 0.90 | 0.003 | 0.0015 |
| CHROME | 70 |  | 0.70 | 0.0014 | 0.0020 |
| DOORS | 2 | 5000 | 1.00 | 0.0016 | 0.0025 |
| JUNK |  |  | -0.05 | 0 | 0.0001 |

### More Linear Programming Models

###### The Disassembly Problem – An Example

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 7.5. Tableau of Jeremiah Junk Yard Example** |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | ESCORTS | 626S | TBIRDS | CADDIES | METAL | SEATS | CHROME | DOORS | JUNK | CONVERT SEATS | CONVERT CHROME | CONVERT DOORS | RHS MIN |
| OBJ | -185 | -210 | -265 | -310 | 0.15 | 0.90 | 0.70 | 1.00 | -0.05 |  |  |  |  |
| METAL | -1380 | -1210 | -1920 | -2418 | 1 |  |  |  |  |  | -1 | -0.7 | = 0 |
| SEATS | -230 | -220 | -192 | -156 |  | 1 |  |  |  | 1 |  |  | = 0 |
| CHROME | -115 | -110 | -288 | -546 |  |  | 1 |  |  |  | 1 |  | = 0 |
| DOORS | -184 | -220 | -320 | -273 |  |  |  | 1 |  |  |  | 1 | = 0 |
| JUNK | -391 | -440 | -480 | -507 |  |  |  |  | 1 | -1 |  | -0.3 | = 0 |
| CAPACITY | 1 | 1 | 1.2 | 1.4 |  |  |  |  |  |  |  |  | ≤ 42 |
| LABOR | 10 | 12 | 15 | 20 | .001 | .0015 | .0020 | .0025 | .0001 |  |  |  | ≤ 700 |
| PART SPACE |  |  |  |  |  | .003 | .0014 | .0016 |  |  |  |  | ≤ 60 |
| LOWER BOUND |  |  |  |  |  | 4000 | 70 | 2 |  |  |  |  |  |
| UPPER BOUND | 13 | 12 | 20 | 10 |  | 6000 | 10000 | 5000 |  |  |  |  |  |

### More Linear Programming Models

###### The Disassembly Problem – An Example

Solution

|  |
| --- |
| **Table 7.6. Solution for Jeremiah Junk Yard Example** |
| Objective = 18337.2 |  |  |  |  |
| Variable |  Value | Reduced Cost |  | Constraint | Slack | Shadow Price |
| ESCORTS | 4.00 | 0 |  | Parts |  |  |
| 626S | 0 | -49.960 |  | METAL | 0 | 0.150 |
| TBIRDS | 20 |  31.688 |  | SEATS | 0 | -0.050 |
| CADDIES | 10 |  91.356 |  | CHROME | 0 | 0.150 |
|  |  |  |  | DOORS | 0 | 0.090 |
| Sell |  |  |  | JUNK | 0 | -0.050 |
| METAL | 73186.2 | 0 |  | CAPACITY | 0 | 24.760 |
| SEATS | 6000  | 0.95 |  | LABOR | 43.512 | 0 |
| CHROME | 10000  | 0.550 |  | PARTSPACE | 20 | 0 |
| DOORS | 5000 | 0.910 |  |  |  |  |
| JUNK | 18013.8  | 0 |  |  |  |  |
|  |  |  |  |  |  |  |
| Convert |  |  |  |  |  |  |
| SEATS | 320 | 0 |  |  |  |  |
| CHROME | 1680 | 0 |  |  |  |  |
| DOORS | 4866 | 0 |  |  |  |  |

### More Linear Programming Models

###### The Assembly-Disassembly Problem

Primal Algebra



Objective: The objective function maximizes the revenue from final products and component parts sold less the costs of the raw products and component parts purchased.

Constraints: The first constraint is a supply‑demand balance, and balances the use of component parts through their assembly into final products and dir­ect sale, with the supply of component parts from either the disassembly oper­ation or purchases.

The remaining equations impose resource limitation con­straints and upper bounds.

### More Linear Programming Models

###### The Assembly-Disassembly Problem

An Example

|  |
| --- |
| **Table 7.7. Data for Chicken Example Yields from Cutting** |
|  | Parts | Halves | Quarters | Meat | Leg-Breast-Thigh |
| Wings | 2 |  |  |  |  |
| Legs | 2 |  |  |  | 2 |
| Thighs | 2 |  |  |  | 2 |
| Back | 1 |  |  |  |  |
| Breasts | 2 |  |  |  | 2 |
| Necks | 1 |  |  |  | 1 |
| Gizzards | 1 | 1 | 1 | 1 |  |
| Meat |  | 0.05 | 0.07 | 1 | 0.2 |
| Breast Quarter |  |  | 2 |  |  |
| Leg Quarter |  |  | 2 |  |  |
| Halves |  | 2 |  |  |  |
| **Selling Price and Labor Use for Chicken Packs**  |
| Pack | Labor | Price |
| A | 2 | $2.05 |
| B | 1.3 |  2.00 |
| C | 1.2 |  1.45 |
| D | 1.1 |  1.95 |
| E | 1.25 |  1.25 |
| Gizzard | 1.0 |  0.90 |
| **Individual Selling Prices for Parts** |
| Part | Price | Part | Price |
| Wings | 0.10 | Gizzards | 0.07 |
| Legs | 0.20 | Meat | 2.00/lb. |
| Thighs | 0.25 | Breast Quarters | 0.45 |
| Backs | 0.12 | Leg Quarter | 0.40 |
| Breasts | 0.33 | Halves | 0.90 |
| Necks | 0.05 |  |  |

### More Linear Programming Models

###### The Assembly-Disassembly Problem

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | B |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Sell |  |  |  | r |  |  | Buy | RHS |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | e |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | G |  | a | L |  |  |  |  |  |  |
|  | Disassemble | Assemble |  |  |  |  | B |  | i |  | s | e | H |  |  | T |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | W |  | T |  | r |  | z |  | t | g | a | W |  | h |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | i |  | h | B | e | N | z | M |  |  | l | i | L | i |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | n | L | i | a | a | e | a | e | Q | Q | v | n | e | g |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | g | e | g | c | s | c | r | a | t | t | e | g | g | h |  |  |
|  | Xp | Xh | Xq | Xm | XL | Xa | Xb | Xc | Xd | Xe | Xg | s | g | h | k | t | k | d | t | r | r | s | s | s | s |  |  |
| Object | -1 | -1 | -1 | -1 | -1 | 2.05 | 2.00 | 1.45 | 1.95 | 1.25 | .90 | .10 | .20 | .25 | .12 | .33 | .05 | .07 | 2.0 | .45 | .40 | .90 | -.12 | -.22 | -.27 | Max |
| Wings | -2 |  |  |  |  | 2 |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  | -1 |  |  | ≤ | 0 |
| Legs | -2 |  |  |  | -2 | 2 |  |  |  | 2 |  |  | 1 |  |  |  |  |  |  |  |  |  |  | -1 |  | ≤ | 0 |
| Thighs | -2 |  |  |  | -2 | 2 |  |  |  | 2 |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  | -1 | ≤ | 0 |
| Backs | -1 |  |  |  |  | 1 |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  | ≤ | 0 |
| Breasts | -2 |  |  |  | -2 | 2 |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  | ≤ | 0 |
| Necks | -1 |  |  |  | -1 | 1 |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  | ≤ | 0 |
| Gizzards | -1 | -1 | -1 | -1 |  |  |  |  |  |  | 10 |  |  |  |  |  |  | 1 |  |  |  |  |  |  |  | ≤ | 0 |
| Meat |  | -.05 | -.07 | -1 | -.2 |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  |  | ≤ | 0 |
| Breast Qtr. |  |  | -2 |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |  | ≤ | 0 |
| Leg Qtr. |  |  | -2 |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | ≤ | 0 |
| Halves |  | -2 |  |  |  |  |  |  | 2 |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  | ≤ | 0 |
| Chickens | 1 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ | 1000 |
| Labor |  |  |  |  |  | 2 | 1.3 | 1.2 | 1.1 | 1.25 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ | 3000 |
| Wing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  | ≤ | 20 |
| Leg |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  | ≤ | 20 |
| Thigh |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 | ≤ | 20 |

**Table 7.8. Primal Formulation of Charles Chicken Company Problem**

### More Linear Programming Models

###### The Assembly-Disassembly Problem

Solution

|  |
| --- |
| **Table 7.9. Solution to the Charles Chicken Co. Problem**  |
| Objective function = 1362.7 |
| Variable | Value | Reduced Cost | Equation | Slack | Shadow Price |
| Xp |  0 | -0.22 | Wings |  0 | 0.120 |
| Xh |  0 | 0 | Legs |  0 |  0.355 |
| Xq |  0 | -0.33 | Thighs |  0 |  0.270 |
| Xm |  0 | -0.27 | Backs |  0 |  0.180 |
| XL |  1000 | 0 | Breasts |  0 |  0.330 |
| Xa |  0 | 0 | Necks |  0 |  0.050 |
| Xb |  0 | 0 | Gizzards |  0 |  0.090 |
| Xc |  0 | -0.15 | Meat |  0 |  2.000 |
| Xd |  0 | -0.22 | Breast Qtr. |  0 |  0.500 |
| Xe |  1010 | 0 | Leg Qtr. |  0 |  0.400 |
| Gizzards |  0 | 0 | Halves |  0 |  1.085 |
| Wings |  0 | -0.02 | Chickens |  0 |  1.36  |
| Legs |  0 | -0.02 | Labor | 1737.5 |  0  |
| Thighs |  0 | -0.155 |  |  |  |
| Backs |  0 | -0.06 |  |  |  |
| Breasts |  2000 | 0 |  |  |  |
| Necks |  1000 | 0 |  |  |  |
| Gizzards |  0 | -0.02 |  |  |  |
| Meat |  200 | 0 |  |  |  |
| Breast Qtr. |  0 | -0.05 |  |  |  |
| Leg Qtr. |  0 | 0 |  |  |  |
| Halves |  0 | -0.185 |  |  |  |
| Wings |  0 | 0 |  |  |  |
| Legs |  20 | 0 |  |  |  |
| Thighs |  20 | 0.135 |  |  |  |

### More Linear Programming Models

###### The Assembly-Disassembly Problem

**Violation of Separability Assumption**

The Blending Problem:



|  |
| --- |
| **Table 7.10. Data for the Grain Blending Example** |
|  | Grade | Characteristics |
|  | Maximums | GrainBatch 1 | GrainBatch 2 |
|  | A | B |  |  |
| Moisture | 1 | 2 | 2 | 1 |
| Foreign Matter | 1 | 2 | 1 | 2 |
| **Table 7.11. Solution of the First Formulation of the Grain Blending Problem** |
| Objective = 100 |
| Variable | Value | Reduced Cost | Equation | Slack | Shadow Price |
| A | 20 | 0 | Moisture | 0 | 1 |
| B | 20 | 0 | Foreign Matter | 0 | 0 |
| G1 | 20 | 2 | Weight | 0 | 4 |
| G2 | 20 | 3 |  |  |  |

There is a problem with this solution. It is impossible, given the data above, to make a mix containing 20 units each of grade A and grade B grain.

### More Linear Programming Models

###### The Assembly-Disassembly Problem

**Violation of Separability Assumption**

The proper formulation of the blending problem is



|  |
| --- |
| **Table 7.12. Optimal Solution to the Correct Formulation of the Grain Blending Problem** |
| Objective = 80 |
| Variable | Value | Reduced Cost | Equation | Slack | Shadow Price |
| A | 0 | 0  | 1 | 0 | 1 |
| B | 40 | 0 | 2 | 0 | 1 |
| G11 | 0 | 0 | 3 | 0 | 5 |
| G12 | 20 | 0 | 4 | 20 | 0 |
| G21 | 0 | 0 | 5 | 20 | 0 |
| G22 | 20 | 0 | 6 | 0 | 2 |
|  |  |  | 7 | 0 | 2 |
|  |  |  | 8 | 0 | 2 |

### More Linear Programming Models

###### Sequencing Problems

Sequencing Constraints:

Assuming that returns and resource usage are independent of activity timing we have:



When returns to the successor activities depend on the timing of the preceding activities we have:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Predecessor date | Wk 1 |  | Wk 1 |  | Wk 1 |  | Wk 2 |  | Wk 2 |  | Wk 3 |  |  |
| Successor date | Wk 1 |  | Wk 2 |  | Wk 3 |  | Wk 2 |  | Wk 3 |  | Wk 3 |  |  |
| Wk 1 | aZ11 | + | bZ12 | + | dZ13 |  |  |  |  |  |  | ≤ | T1 |
| Wk 2 |  |  | cZ12 |  |  | + | fZ22 | + | gZ23 |  |  | ≤ | T2 |
| Wk 3 |  |  |  |  | eZ13 |  |  | + | hZ23 | + | iZ33 | ≤ | T3 |

### More Linear Programming Models

###### Sequencing Problems

###### General Formulation



### More Linear Programming Models

###### Sequencing Problems- Example 1

|  |
| --- |
| **Table 7.13. LP Formulation of Sequencing Example 1** |
|  |  | Plow - X | Disc - Y | Plant etc. - Z | RHS |
|  |  | April | May | June | May | June | July | May | June | July |  |  |
| Obj |  | -100 | -100 | -100 | -20 | -20 | -20 | 400 | 400 | 400 | max |
| X – Y | May | -1 | -1 |  | 1 |  |  |  |  |  | ≤ | 0 |
| link | June | -1 | -1 | -1 | 1 | 1 |  |  |  |  | ≤ | 0 |
|  | July | -1 | -1 | -1 | 1 | 1 | 1 |  |  |  | ≤ | 0 |
| Y – Z | May |  |  |  | -1 |  |  | 1 |  |  | ≤ | 0 |
| link | June |  |  |  | -1 | -1 |  | 1 | 1 |  | ≤ | 0 |
|  | July |  |  |  | -1 | -1 | -1 | 1 | 1 | 1 | ≤ | 0 |
| Labor | April | 0.2 |  |  |  |  |  |  |  |  | ≤ | 160 |
|  | May |  | 0.2 |  | 0.3 |  |  | 0.3 |  |  | ≤ | 160 |
|  | June |  |  | 0.2 |  | 0.3 |  | 0.1 | 0.3 |  | ≤ | 160 |
|  | July |  |  |  |  |  | 0.3 | 0.1 | 0.1 | 0.3 | ≤ | 160 |
|  | Aug. |  |  |  |  |  |  | 0.1 | 0.1 | 0.1 | ≤ | 160 |
|  | Sept. |  |  |  |  |  |  | 0.5 | 0.1 | 0.1 | ≤ | 160 |
|  | Oct. |  |  |  |  |  |  |  | 0.5 | 0.1 | ≤ | 160 |
|  | Nov. |  |  |  |  |  |  |  |  | 0.5 | ≤ | 160 |
| Land |  | 1 | 1 | 1 |  |  |  |  |  |  | ≤ | 600 |

### More Linear Programming Models

###### Sequencing Problems-Example 1 Solution

|  |  |
| --- | --- |
|  | **Table 7.14. Solution to Sequencing Example 1** |
|  | Objective function = 168,000 |
| Variable | Value | Reduced Cost |  | Equation | Slack | Shadow Price |
| Plow | April | 600 | 0 | Plow-Disc |  May | -192.59 | 0 |
|  | May | 0 |  0 (alt) |  | June | 200.00 | 0 |
|  | June | 0 |  0 (alt) |  | July | 0 | 380 |
| Disc | May | 407.41 | 0 | Disc-Plant | May | 88.89 |  0  |
|  | June | 0 | 0 |  | June | 0 |  0  |
|  | July | 192.59 | 0 |  | July | 0 | 400 |
| Plant | May | 125.93 | 0 | Labor | April | 97.78 | 0 |
|  | June | 281.48 | 0 |  | May | 0 |  0  |
|  | July | 192.59 | 0 |  | June | 0 | 0 |
|  |  |  |  |  | July | 0 |  0  |
|  |  |  |  |  | Aug. | 100 | 0 |
|  |  |  |  |  | Sept. | 11.11 | 0 |
|  |  |  |  |  | Oct. | 51.11 | 0 |
|  |  |  |  |  | Nov. | 60 | 0 |
|  |  |  |  | Land |   | 0 | 280 |

### More Linear Programming Models

###### Sequencing Problems-Example 2

This example reflects a farm planning situa­tion and illustrates what needs to be done when planting and harvesting date influence yield

|  |
| --- |
| **Table 7.15. Yields for Crops 1 and 2 by Crop Planting and Harvest Dates** |
|  | Planting Date |
| HarvestDate | Crop 1 | Crop 2 |
|  | April | May | June | April | May | June |
| September | 110 | 105 | 90 | 38 | 40 | 35 |
| October | 125 | 120 | 118 | 35 | 38 | 40 |

### More Linear Programming Models

###### Sequencing Problems-Example 2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Rows |  |  |  |  |  |  |  |  |  |  |  |  |  | Mar | April | May | Mar | April | May |  |  |  |  |  |  |  |  |  |   |
|  | Mar | Apr | May | Jun | Apr | May | Jun | Apr | May | Jun | Apr | May | Jun | Apr | May | Jun | Apr | May | Jun | Mar | Apr | May | Jun | Sep | Oct | Nov | Crop 1 | Crop 2 |   |
|  |  |  |  |  |  |  |  | Sep | Sep | Sep | Oct | Oct | Oct | Sep | Sep | Sep | Oct | Oct | Oct |  |  |  |  |  |  |  |  |  |   |
| Objective | -5 | -5 | -5 | -3 | -3 | -3 | -60 | -60 | -60 | -60 | -60 | -60 | -43 | -43 | -43 | -43 | -43 | -43 | -10 | -10 | -10 | -10 | -10 | -10 | -10 | -10 | 3 | 8.7 | Max |
| Land Balance | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ 1500 |
|  | Mar | -1 |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Plowed | Apr | -1 | -1 |  |  | 1 |  |  |  |  |  |  |  |  | 1 | 1 |  | 1 | 1 |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Land | May | -1 | -1 | -1 |  | 1 | 1 |  |  |  |  |  |  |  | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Balan | Jun | -1 | -1 | -1 | -1 | 1 | 1 | 1 |  |  |  |  |  |  | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Disced | Apr |  |  |  |  | -1 |  |  | 1 |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Land | May |  |  |  |  | -1 | -1 |  | 1 | 1 |  | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
| Balan | Jun |  |  |  |  | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ≤ 0 |
|  | Mar | 0.3 |  |  |  |  |  |  |  |  |  |  |  |  | 0.2 |  |  | 0.2 |  |  | -1 |  |  |  |  |  |  |  |  | ≤ 300 |
|  | Apr |  | 0.3 |  |  | 0.2 |  |  | 0.22 |  |  | 0.22 |  |  | 0.22 | 0.2 |  | 0.22 | 0.2 |  |  | -1 |  |  |  |  |  |  |  | ≤ 300 |
| Labor | May |  |  | 0.3 |  |  | 0.2 |  | 0.1 | 0.22 |  | 0.1 | 0.22 |  | 0.1 | 0.22 | 0.2 | 0.1 | 0.22 | 0.2 |  |  | -1 |  |  |  |  |  |  | ≤ 300 |
| Avail- | Jun |  |  |  | 0.3 |  |  | 0.2 |  | 0.1 | 0.22 |  | 0.1 | 0.22 |  | 0.1 | 0.22 |  | 0.1 | 0.22 |  |  |  | -1 |  |  |  |  |  | ≤ 300 |
| Ability | Jul |  |  |  |  |  |  |  |  |  | 0.1 |  |  | 0.1 |  |  | 0.1 |  |  | 0.1 |  |  |  |  | -1 |  |  |  |  | ≤ 300 |
|  | Sep |  |  |  |  |  |  |  | 0.7 | 0.7 | 0.7 |  |  |  | 0.6 | 0.6 | 0.6 |  |  |  |  |  |  |  |  | -1 |  |  |  | ≤ 300 |
|  | Oct |  |  |  |  |  |  |  |  |  |  | 0.7 | 0.7 | 0.7 |  |  |  | 0.6 | 0.6 | 0.6 |  |  |  |  |  |  | -1 |  |  | ≤ 300 |
| Yield | Crop 1 |  |  |  |  |  |  |  | -110 | -105 | -90 | -125 | -120 | -118 |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  | ≤ 0 |
|  | Crop 2 |  |  |  |  |  |  |  |  |  |  |  |  |  | -38 | -40 | -35 | -35 | -38 | -40 |  |  |  |  |  |  |  |  | 1 | ≤ 0 |

### More Linear Programming Models

###### Sequencing Problems-Example 2 Solution

|  |
| --- |
| **Table 7.17. Solution for Sequencing Example 2** |
| Objective function = 449,570 |
| Variable |  | Value | Reduced Cost | Equation |  | Slack | Shadow Price |
| Acreage Plowed in: | March | 1275 | 0 | Land |  | 0 | 292.5 |
|  | April | 0 | 0 | Plowed Land: | March | 1275 | 0 |
|  | May | 225 | 0 |  | April | 0 | 2.10 |
|  | June | 0 | 0 |  | May | 0 | 14.4 |
| Acreage Disced for Crop 1 in: | April | 775 | 0 |  | June | 0 | 284.0 |
|  | May | 0 | 0 | Disced Land: | April | 0 | 13.16 |
|  | June | 0 | 0 |  | May | 0 | 5.34 |
| Acreage of Crop 1 planted/harvested in: | Sept./April | 0 | -40.15 |  | June | 0 | 287.0  |
|  | Sept./May | 0 | -49.81 | Labor: | March | 0 | 10 |
|  | Sept./June | 0 | -92.65 |  | April | 0 | 10 |
|  | Oct./April | 775 | 0 |  | May | 0 | 3 |
|  | Oct./May | 0 | -9.66 |  | June | 200.5 | 0 |
|  | Oct./June | 0 | -13.5 |  | July | 277.5 | 0 |
| Acreage of Crop 2 planted/harvested in: | Sept./April | 0 | -19.24 |  | Sept. | 0 | 3.067 |
|  | Sept./May | 500 | 0 |  | Oct. | 0 | 10 |
|  | Sept./June | 0 | -39.34 | Yield: | Crop 1 | 0 | 3 |
|  | Oct./April | 0 | -49.5 |  | Crop 2 | 0 | 8.7 |
|  | Oct./May | 0 | -21.56 |  |  |  |  |
|  | Oct./June | 225 | 0 |  |  |  |  |
| Labor hired in: | March | 82.5 | 0 |  |  |  |  |
|  | April | 125.5 | 0 |  |  |  |  |
|  | May | 0 | -7 |  |  |  |  |
|  | June | 0 | -10 |  |  |  |  |
|  | July | 0 | -10 |  |  |  |  |
|  | Sept. | 0 | -6.93 |  |  |  |  |
|  | Oct. | 377.5 | 0 |  |  |  |  |
| Crop 1 Sales |  | 96875 | 0 |  |  |  |  |
| Crop 2 Sales |  | 29000 | 0 |  |  |  |  |

### More Linear Programming Models

###### The Storage Problem

**Primal Algebra**



Objective: It involves summation across all the periods of the revenues from the sales of the good less the costs of storage of the good. We only include storage from the time periods 1 through T‑1, assuming that everything must be sold in the last time period.

Constraints: The first constraint limits the quantity sold in the first period plus the quantity stored into the second period to be less than or equal to the initial inventory available.

The next con­straints are active in all time periods excepting 1 and T. This limits the amount sold in each period plus the amount stored into the next period to not exceed the amount held over from the period before.

The third constraint gives the inventory con­dition for the last time period requiring that sales not exceed inventory carried over from the time period before.

The next two constraints impose upper and lower limits on the amount that can be sold during any time period.

The last constraint imposes an upper limit on storage in the first period.

### More Linear Programming Models

###### The Storage Problem – An Example

|  |  |
| --- | --- |
|  | **Table 7.18. Formulation of Storage Example** |
| Objective |  |  | Sell |  |  |  |  |  |  | Store |  |  |  |  |
|  | 2.3X1 | + | 2.5X2 | + | 2.7X3 | + | 2.9X4 | - | .1h1 | - | .2h2 | - | .3h3 |  |  |
| Grain Inventory |  1 | X1 |  |  |  |  |  |  | + | h1 |  |  |  |  | ≤ | 100 |
|  |  2  |  |  | X2 |  |  |  |  | - | h1 | + | h2 |  |  | ≤ | 0  |
|  |  3 |  |  |  |  | X3 |  |  |  |  | - | h2 | + | h3 | ≤ | 0 |
|  |  4 |  |  |  |  |  |  | X4 |  |  |  |  | - | h3 | ≤ | 0 |
|  |  1 | X1 |  |  |  |  |  |  |  |  |  |  |  |  | ≤ | 50 |
| Max |  2 |  |  | X2 |  |  |  |  |  |  |  |  |  |  | ≤ | 50 |
| Sales |  3 |  |  |  |  | X3 |  |  |  |  |  |  |  |  | ≤ | 50 |
|  |  4 |  |  |  |  |  |  | X4 |  |  |  |  |  |  | ≤ | 50 |
| Min |  1 | X1 |  |  |  |  |  |  |  |  |  |  |  |  | ≥ | 15 |
| Sales |  2 |  |  | X2 |  |  |  |  |  |  |  |  |  |  | ≥ | 5 |
| Max Store |  |  |  |  |  |  |  |  | h1 |  |  |  |  | ≤ | 75 |

### More Linear Programming Models

###### The Storage Problem – Example Solution

|  |
| --- |
| **Table 7.19. Primal Solution to the Storage Problem Example** |
| Objective = 237.5 |
| Variable | Value | Reduced Cost | Constraint | Slack | Shadow Price |
| X1 | 25 | 0 | Pd1 Inventory | 0 | 2.3 |
| X2 | 50 | 0 | Pd2 Inventory | 0 | 2.5 |
| X3 | 25 | 0 | Pd3 Inventory | 0 | 2.7 |
| X4 | 0 | 0 | Pd4 Inventory | 0 | 2.9 |
| h1 | 75 | 0 | Max sale Pd1 | 25 | 0 |
| h2 | 25 | 0 | Max sale Pd2 | 0 | 0 |
| h3 | 0 | -0.1 | Max sale Pd3 | 25 | 0 |
|  |  |  | Max sale Pd4 | 50 | 0 |
|  |  |  | Capacity | 0 | 0.1 |
|  |  |  | Min sale Pd1 | 10 | 0 |
|  |  |  | Min sale Pd2 | 45 | 0 |
|  |  |  | Min sale Pd3 | 25 | 0 |
|  |  |  | Min sale Pd4 | 0 | 0 |

### More Linear Programming Models

###### Input-Output Analysis













### More Linear Programming Models

###### Input-Output Analysis – An Example

|  |
| --- |
| **Table 7.20. Input Output Example Data** |
|  | Transactions Matrix |
|  | Manufacturing | Agriculture | Finance | Services |
| Manufacturing | 50 | 40 | 10 | 75 |
| Agriculture | 20 | 10 | 2 | 40 |
| Finance | 25 | 8 | 12 | 20 |
| Services | 100 | 40 | 40 | 40 |
| Exogenous | 55 | 24 | 11 | 55 |

|  |
| --- |
| **Final Demand Data** |
| Sector | Final Demandfor Sectors |
| Manufacturing | 75 |
| Agriculture | 50 |
| Finance | 10 |
| Services | 10 |

|  |
| --- |
| **Table 7.21. Technical Coefficient Matrix for Input Output**  |
|  | Manufacturing | Agriculture | Finance | Services |
| Manufacturing | 0.200 | 0.328 | 0.133 | 0.326 |
| Agriculture | 0.080 | 0.082 | 0.027 | 0.174 |
| Finance | 0.100 | 0.066 | 0.160 | 0.087 |
| Services | 0.400 | 0.328 | 0.533 | 0.174 |
| Exogenous | 0.220 | 0.197 | 0.147 | 0.239 |

### More Linear Programming Models

###### Input-Output Analysis – An Example

**Empirical Setup**

|  |
| --- |
| **Table 7.22. LP Formulation of Input Output Example**  |
|  | Manufacturing | Agriculture | Finance | Services |  |
| Maximize | 1 | 1 | 1 | 1 |  |
| Manufacturing | 0.8 | -0.33 | -0.13 | -0.33 | ≤ 75 |
| Agriculture | -0.08 | 0.92 | -0.03 | -0.17 | ≤ 50 |
| Finance | -0.1 | -0.07 | 0.84 | -0.09 | ≤ 10 |
| Services | -0.4 | -0.33 | -0.53 | 0.83 | ≤ 10 |

**Solution**

|  |
| --- |
| **Table 7.23. Solution for Input Output Example** |
| Objective = 677 |  |  |  |  |
| Variable | Value | Reduced Cost | Constraint | Slack | Shadow Price |
| Manufacturing | 250 | 0 | Manufacturing | 0 | 4.615 |
| Agriculture | 122 | 0 | Agriculture | 0 | 4.716 |
| Finance | 75 | 0 | Finance | 0 | 4.960 |
| Services | 230 | 0 | Services | 0 | 4.547 |

### More Linear Programming Models

###### Block Diagonal

* This model depicts production in several different locations and/or time periods.
* The blocks arise when individual production units utilize immobile resources.
* The problem also depicts some usage of unifying resources at the overall firm level.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Max |  | + |  |  |  |  |
|  | s.t. |  | + |  | ≤ | bi |  for all I |
|  |  |  |  |  | ≤ | fLM |  for all L and M |
|  |  | Xk | , | YjL | ≥ | 0 |  for all k, j and L |

**Objective**: The problem maximizes profit summed over the global and sub‑unit activities sub­ject to an overall linking constraint and individual sub‑unit constraints.

**A Closer Look**



### More Linear Programming Models Block Diagonal - Example

|  |
| --- |
| **Table 7.24. Matrix Formulation of Block Diagonal Problem** |
|  |  | PLANT 1 | PLANT 2  | PLANT 3 |  |  |
|  |  | Sell Sets FC FY | MakeTable FC FY | Sell Table  | Transport ChairFC FY | Sell Chair FC FY | Make Functional Chairs Norm MxSm MxLg | Make FancyChairs Norm MxSm MxLg | Transport Table FC FY | Transport Chair FC FY | Sell Table FC FY | Sell Chair FC FY | Make TableFC FY | Make Functional Chairs Norm MxSm MxLg | Make Fancy ChairsNorm MxSm MxLg | RHS |
| Objective | 600 100 | -80 -100 | 200 300 | -5 -5 | 82 105 | -15 -16 -15.7 | -25 -26 -26.6 | -20 -20 | -7 -7 | 200 300 | 82 105 | -80 -100 | -15 -16 -15.7 | -25 -26.5 -26.5 | Max |
| PLANT1 | Table FC |  1 | -1 |  1 |  |  |  |  | -1 |  |  |  |  |  |  | ≤ | 0 |
| Inventory FY | 1  | -1  | 1 |  |  |  |  | -1 |  |  |  |  |  |  | ≤ | 0 |
| Chair FC |  4 |  |  | -1 |  |  |  |  | -1 |  |  |  |  |  | ≤ | 0 |
| Inventory FY | 6  |  |  |  -1  |  |  |  |  | -1 |  |  |  |  |  | ≤ | 0 |
| Labor  |  | 3 |  5 |  |  |  |  |  |  |  |  |  |  |  |  | ≤ | 175 |
| Top Capacity |  | 1 |  1 |  |  |  |  |  |  |  |  |  |  |  |  | ≤ | 50 |
| P L ANT2 | Chair FC |  |  |  |  1 | 1 | -1 -1 -1 |  |  |  |  |  |  |  |  | ≤ | 0 |
| Inventory FY |  |  |  | 1  | 1 |  | -1 -1 -1 |  |  |  |  |  |  |  | ≤ | 0 |
| Small Lathe |  |  |  |  |  | 0.8 1.3 0.2 | 1.2 1.7 0.5 |  |  |  |  |  |  |  | ≤ | 140 |
| Large Lathe |  |  |  |  |  | 0.5 0.2 1.3 | 0.7 0.3 1.5 |  |  |  |  |  |  |  | ≤ | 90 |
| Chair Bottom Carver |  |  |  |  |  | 0.4 0.4 0.4 |  1 1 1  |  |  |  |  |  |  |  | ≤ | 120 |
| Labor |  |  |  |  |  | 1 1.05 1.1 | 0.8 0.82 0.84 |  |  |  |  |  |  |  | ≤ | 125 |
| PLN3 | Table FC |  |  |  |  |  |  |  | 1 |  | 1 |  | -1 |  |  | ≤ | 0 |
| Inventory FY |  |  |  |  |  |  |  | 1 |  | 1 |  | -1 |  |  | ≤ | 0 |
| Chair FC |  |  |  |  |  |  |  |  | 1 |  | 1 |  | -1 -1 -1 |  | ≤ | 0 |
|  | Inventory FY |  |  |  |  |  |  |  |  | 1 |  | 1 |  |  | -1 -1 -1 | ≤ | 0 |
|  | Small Lathe |  |  |  |  |  |  |  |  |  |  |  |  | 0.8 1.3 0.2 | 1.2 1.7 0.5 | ≤ | 130 |
|  | Large Lathe |  |  |  |  |  |  |  |  |  |  |  |  | 0.5 0.2 1.3 | 0.7 0.3 1.5 | ≤ | 100 |
|  | Chair Bottom Carver |  |  |  |  |  |  |  |  |  |  |  |  | 0.4 0.4 0.4 | 1 1 1 | ≤ | 110 |
|  | Labor |  |  |  |  |  |  |  |  |  |  |  | 3 5 | 1 1.05 1.1 | 0.80 0.82 0.84 | ≤ | 210 |
|  | Top Capacity |  |  |  |  |  |  |  |  |  |  |  | 1 1 |  |  | ≤ | 40 |

### More Linear Programming Models

###### Block Diagonal – Example Solution

|  |
| --- |
| Table 7.25. Primal Solution to the Block Diagonal Problem |
| Objective = 36206.9 |
| Variable |  | Value | Reduced Cost | Equation  | Slack | Shadow Price |
| Plant1 | Sell FC set | 24.40 | 0 | Plant1 | FC Tables | 0 | 212 |
|  | Sell FY set | 29.01 | 0 |  | FY Tables | 0 | 320 |
|  | Make FC Table | 24.40 | 0 |  | FC Chairs | 0 | 97 |
|  | Make FY Table | 20.36 | 0 |  | FY Chairs | 0 | 130 |
|  | Sell FC Table | 0 | -12 |  | Labor | 0 | 44 |
|  | Sell FY Table | 0 | -20 |  | Top Cap | 5.240 | 0 |
| Plant2 | Trans FC Chair | 62.23 | 0 | Plant2 | FC Chair | 0 | 92 |
|  | Trans FY Chair | 78.2 | 0 |  | FY Chair | 0 | 125 |
|  | Sell FC Chair | 0 | -10 |  | Sm Lathe | 0 | 47.77 |
|  | Sell FY Chair | 0 | -20 |  | Lrg Lathe | 0 | 38.83 |
|  | Make FC Table | 0 | -58.11 |  | Chair Bot | 16.907 | 0 |
|  | Make FY Table | 0 | -96.85 |  | Labor | 0 | 19.37 |
|  | Make FC Chair N | 62.23 | 0 | Plant3 | FC Table | 0 | 200 |
|  | Make FC Chair MS | 0 | -14.2 |  | FY Table | 0 | 300 |
|  | Make FC Chair ML | 0 | -5.04 |  | FC Chair | 0 | 90 |
|  | Make FY Chair N | 73.02 | 0 |  | FY Chair | 0 | 123 |
|  | Make FY Chair MS | 0 | -10.24 |  | Sm Lathe | 0 | 18.50 |
|  | Make FY Chair ML | 5.18 | 0 |  | Lrg Lathe | 0 | 12.19 |
| Plant3 | Trans FC Table | 0 | -8 |  | Chair Bot | 0 | 35.27 |
|  | Trans FY Table | 8.649 | 0 |  | Labor | 0 | 40.00 |
|  | Trans FC Chair | 35.37 | 0 |  | Top Cap | 20.562 | 0 |
|  | Trans FY Chair | 95.85 | 0 |  |  |  |  |
|  | Sell FC Table  | 0 | 0 |  |  |  |  |
|  | Sell FY Table | 10.79 | 0 |  |  |  |  |
|  | Sell FC Chair | 0 | -8 |  |  |  |  |
|  | Sell FY Chair | 0 | -18 |  |  |  |  |
|  | Make FC Table | 0 | 0 |  |  |  |  |
|  | Make FY Table | 19.44 | 0 |  |  |  |  |
|  | Make FC Chair N | 35.37 | 0 |  |  |  |  |
|  | Make FC Chair MS | 0 | -8.59 |  |  |  |  |
|  | Make FC Chair ML | 0 | -3.35 |  |  |  |  |
|  | Make FY Chair N | 76.83 | 0 |  |  |  |  |
|  | Make FY Chair MS | 0 | -6.68 |  |  |  |  |
|  | Make FY Chair ML | 19.02 | 0 |  |  |  |  |

###