Personal scheduling for cash register employees in a retail chain supermarket

Juan Carlos Rodriguez Noriega

Workforce Scheduling Problems

- *Rotating workforce schedules. (Rostering.)*
- *Noncyclic workforce schedules. (ETP)*

• Formal definition of ETPs

Requirements: Each shift S_j is composed of a number of tasks, some of them multiple times.

Ability: Each employee has qualifications that enable him to fulfill certain types of tasks; that is, each employee E_i has a set of tasks ¶T_{i1}, . . . , T_{ir}♢ that E_i can be assigned to.

Availability: There are personal preferences of employees which restrict them to be assigned only to subsets of the shifts.

Conflicts: Obviously, an employee cannot be assigned to more than one task in the same shift. In addition, employees cannot be assigned to two shifts that are in conflict with each other

• PL model

$$find \quad x_{ijk} \quad such \quad that$$

$$\sum_{i=1}^{m} x_{ijk} = R_{jk} \qquad (j = 1..n; k = 1..p) \qquad (2.1)$$

$$x_{ijk} \le Q_{ik} \qquad (i = 1..m; j = 1..n; k = 1..p) \qquad (2.2)$$

$$x_{ijk} \le A_{ij} \qquad (i = 1..m; j = 1..n; k = 1..p) \qquad (2.3)$$

$$\sum_{k=1}^{p} x_{ij_1k} \cdot \sum_{k=1}^{p} x_{ij_2k} \le c_{j_1j_2i} \qquad (i = 1..m; j_1, j_2 = 1..n) \qquad (2.4)$$

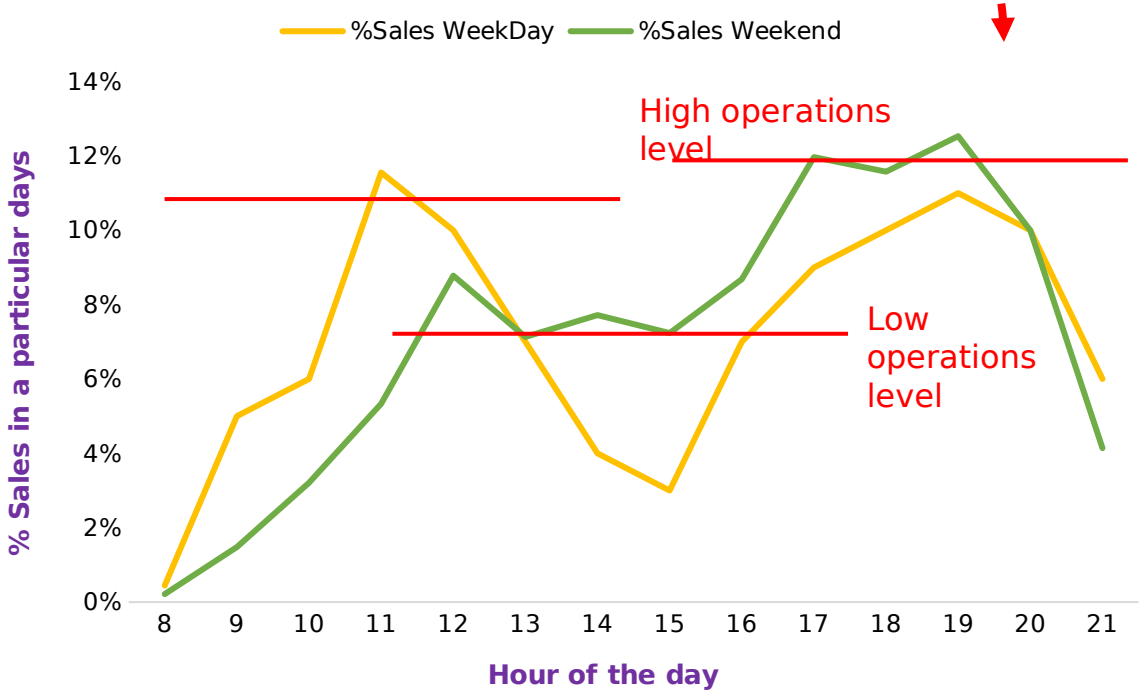
$$V_{ih} \le \sum_{j \in G_h} \sum_{k=1}^{p} x_{ijk} \le W_{ih} \qquad (i = 1..m; h = 1..s) \qquad (2.5)$$

Workload: There is an upper and lower limit on the number of tasks that each employee can be assigned to, per schedule.

Source:Kaplansky, E., & Meisels, A. (2007). Distributed personnel scheduling—negotiation among scheduling agents. *Annals of Operations Research*, *155*(1), 227-255.

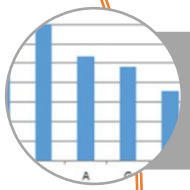
Problem ?

What is the best employee scheduling for cash register that *reduce the cost*, so that we can satisfy the expected demand, respecting the *employee availability* ?



Problem considerations

Employees can do different task (Polyvalent workers)



Demand desegregations per hours follows a patterns for weekdays and weekends.

Each employee must sales 1000K (COP) per hours.

One employee can not work all weekends.

Methodology solution for the problem.

Data reading

Determining the number of employee per hour.

Determine the optimal number of employees in each work shift per day.

Assign to each employee a determinate work shift per day.

Out put solutions

Data input

Work shift: Indicate the time served by each work shift.

	Hour of the day					
Shift	H8	H9	H10	H11	H12	H13
S1	1	1	1	1	0	1
S2	0	1	1	1	1	0
S3	0	0	1	1	1	1

Availability: Indicate what work shift each employee can make.

	Work Shift					
Employee	S1	S2	S3	S4	S5	S6
1	1	1	1	1	1	1
2	1	1	1	1	0	1
3	1	1	1	1	1	1

Sales behavior: Indicate the % of sales by hours for weekdays and weekends.

Hour	%Sales WeekDay	%Sales Weekend
8	0%	0%
9	5%	1%
10	6%	3%

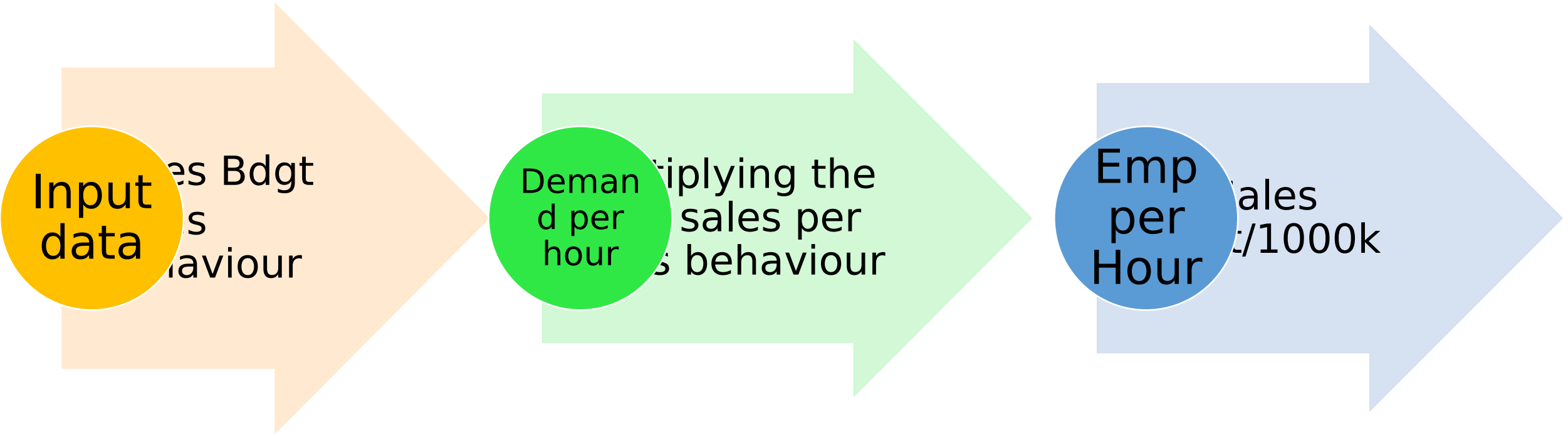
Demand: Sales by day (in 1000k COP).

Dia	Date	Week	Week Day	Sales
1	04/03/2018	1	domingo	$ 44,889
2	05/03/2018	1	lunes	$ 17,585
3	06/03/2018	1	martes	$ 20,017

WS cost: Total cost of a WS (in 1000k COP)

Shift	Shift Cost
S1	40.00
S2	40.00
S3	40.00

Determining the number of employee per hour.



Determine the optimal number of employees in each work shift per day.

Mathematical model : *Determining the work shift par day*

- ***Phase 1:***

n : Number of work shift
nc : Number of work shift full time
m : Number of working hours
Aij : Work shift constant matrix (n x m).
Dj : Number of employee required in hour j (n x m).
Ci : Cost of selecting work shift i.

Xi : Number of employee to be assigned in work shift i.

Model ran for each day in planning horizon

MINIMIZE $\sum_{i \in n} C_i * X_i$

1-Minimize the total cost for scheduling in a day

SUBJECT TO

$$\sum_{i \in n} A_{ij} * X_i \geq D_j ; \forall j \in m$$

2-De demand of employee must be satisfied in a specific hour.

$$\sum_{i \in n_c} X_i \geq 1$$

3-At least one full-time work shift must be assigned

$$X_i \geq 0$$

4-Not negativity constraint

Result for the Phase 1 LP model

		Day hour														
WS	#Emplo Assigned	8	9	10	11	12	13	14	15	16	17	18	19	20	21	Cost WS
S1	1	1	1	1	1	0	1	1	1	1	0	0	0	0	0	$ 40
S2	1	0	1	1	1	1	0	1	1	1	1	0	0	0	0	$ 40
S3	1	0	0	1	1	1	1	0	1	1	1	1	0	0	0	$ 40
S4	1	0	0	0	1	1	1	1	0	1	1	1	1	0	0	$ 40
S5	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	$ 40
S6	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	$ 40
S7	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	$ 40
S8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$ -
S17	2	0	0	0	0	0	0	0	0	0	2	2	2	2	0	$ 40
S18	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	$ 20
															TC	$ 340

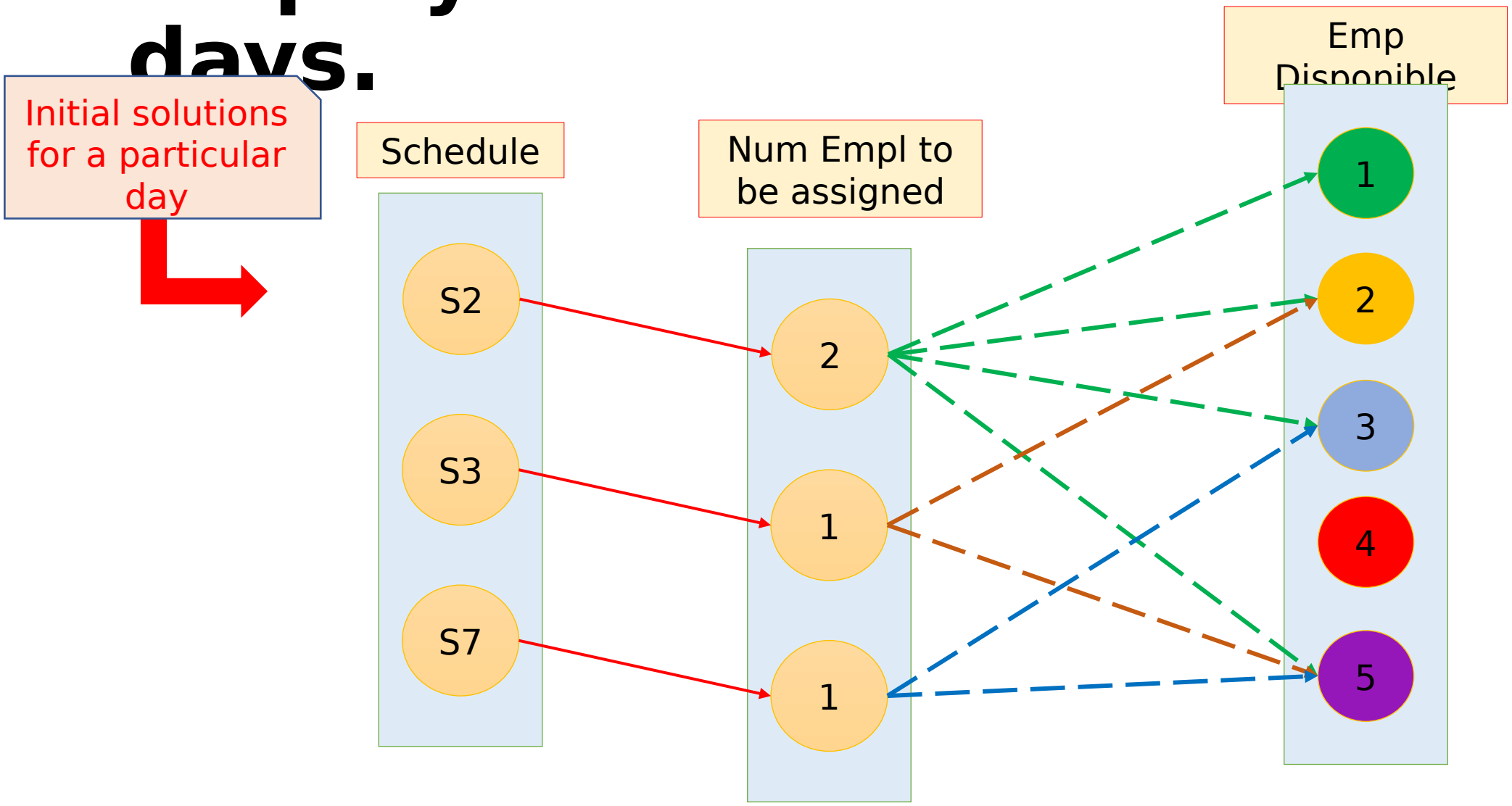
#Employee required	1	1	2	3	4	4	4	4	4	6	6	6	5	2

#Employee assigned	1	2	3	4	4	5	6	6	7	8	8	7	5	2
#Etra employ	0	1	1	1	0	1	2	2	3	2	2	1	0	0

For each day, we determine how many employee must be assigned to each WS, so the cost of the planning for a specific day is minimum

Assign to each employee a determinate work shift per day.

Problem of assigning different employees to the work shift by days.



Mathematical model: Assigning the employee to the work shift

- ***Phase 2:***

n : Number of employee
m : Number of work shift (in a particular day.)
p : Number of days
pc : weekend days
Soljk : solution schedule 'I' in day j (m x p)
αij : Employee availability (n x m).

Yijk : Take the value of 1, if employee i is assigned to work shift j in day k, 0 otherwise

MAXIMIZE $\sum_{i \in n} \sum_{j \in m} \sum_{k \in p} Y_{ijk}$

1-Minimize the total employee assigned to the different work shift

SUBJECT TO

$$\sum_{i \in n} Y_{ijk} = Sol_{jk}; \forall j \in m; \forall k \in p$$

2-The number in the employee to the different work shift must be equal to the initial solution

$$\sum_{j \in m} Y_{ijk} \leq 1; \forall i \in n; \forall k \in p$$

3-One employee must be assigned one time to a work shift in a particular day

$$Y_{ijk} \leq M * \alpha_{ij}; \forall i \in n; \forall k \in p$$

4-Every assigned work shift must respect the employee disponibility.

$$\sum_{k \in pc} Y_{ijk} \leq 5; \forall i \in n; \forall j \in m$$

5-Each employee has to work a maximum of half the number of weekends available.

$$Y_{ijk} \in \{0,1\}$$

6-Not negativity constraint

Output solution

	Day of the month							
	1	2	3	4	5	6	7	8
Employee	domingo	lunes	martes	miércoles	jueves	viernes	sábado	domingo
0	0	0	1	1	1	1	0	0
1	1	0	0	0	0	0	1	1
2	0	0	0	0	0	0	0	0
3	1	0	0	0	0	0	1	1
4	0	0	0	0	0	0	0	0
5	0	1	1	1	1	1	0	1
6	1	1	1	1	1	1	0	0
7	0	1	1	0	0	0	0	1
8	1	0	1	0	0	0	0	1

Proposed solution for day 1

	Hour													
Employee	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20	H21
1	0	0	0	1	1	1	1	0	1	1	1	1	0	0
3	0	0	0	1	1	1	1	0	0	0	0	0	0	0
6	0	0	1	1	1	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	1	1	1	1	0
9	0	0	0	0	1	1	1	1	0	0	0	0	0	0
10	0	0	0	0	0	0	1	1	1	1	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	1	1	1	0
12	0	0	0	0	0	0	0	1	1	1	1	0	0	0
13	0	0	0	0	0	0	0	1	1	1	1	0	0	0
14	1	1	1	1	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	1	1	1	1
16	0	0	0	0	0	0	0	0	0	0	1	1	1	1
18	0	0	0	0	0	0	0	0	0	1	1	1	1	0
#Empl Planified	*1*	*1*	*2*	*4*	*4*	*4*	*4*	*4*	*4*	*7*	*8*	*6*	*5*	*2*
#Empl required	*1*	*1*	*2*	*3*	*4*	*4*	*4*	*4*	*4*	*6*	*6*	*6*	*5*	*2*
#Plan - Req	*0*	*0*	*0*	*1*	*0*	*0*	*0*	*0*	*0*	*1*	*2*	*0*	*0*	*0*

Thank you!