APPLICATION OF ASSIGNMENT PROBLEM THEORY TO CAL POLY'S WEEK OF WELCOME PROGRAM PRESENTATION SCHEDULE

by

KEVIN T. ABAN

A Senior Project submitted

in partial fulfillment

of the requirements for the degree of

Bachelor of Science in Industrial Engineering

California Polytechnic State University

San Luis Obispo

Graded By:__________________________ Date of Submission: _____March 25, 2013________

Abstract

The Week of Welcome (WOW) is a part of Cal Poly's Orientation Programs. One of the main objectives of WOW is to provide content to new students on a variety of different topics in the form of Awareness Presentations. The problem being addressed in this paper is that of assigning the orientation groups of incoming students to these Awareness Presentations so that they have a schedule that meets all of the necessary criteria. The most important of those requirements is schedule balance and flexibility for future use. This scheduling problem is solved through the use of operations research and binary integer programming. Since the times and locations of the presentations for next year's WOW are currently undecided, this project uses past data from last year's WOW to build a model that will serve as the foundation for future years. This paper details several solutions to last year's presentation schedule and conducts comparative analysis among the proposed solutions, as well as the existing process.

Table of Contents

Introduction

The Week of Welcome Program has been an integral part of life at California Polytechnic State University (Cal Poly) for over 50 years. Originating in 1957 as "Welcome Week", it has since become one of the University's oldest traditions, and one that has helped to distinguish Cal Poly and make it unique. Known to all involved as WOW, nearly every student, alumni, and faculty has at one time participated in the program in some way, and helped it slowly evolve into what it is today. On Cal Poly's Student Life and Leadership Page, WOW's goals have been summarized in five ways:

1. WOW introduces the new students to the community and businesses around them.
2. WOW gives students an opportunity to prepare for their classes and meet their advisors.
3. WOW introduces new students to important issues they may encounter while at this university.
4. WOW provides a social atmosphere in which new students will make lasting relationships at their new school.
5. WOW allows new students to experience a taste of college life before classes actually begin.

These goals have been accomplished again and again, year after year, allowing the program to gain a prestigious reputation. That reputation being that WOW has been the perfect means to help incoming students seamlessly transition into college life at Cal Poly. No other university comes to close to matching Cal Poly's effort in Orientation Programs. To outsiders, the structure of WOW is simple – two leaders guide a group of approximately 10 to 15 students for five days, taking them to presentations, to meals, to recreation; to everything a Cal Poly student might do

during their stay in San Luis Obispo. But to those involved, it becomes much more. Most of the leaders spent about 100 hours preparing and being a part of the program. Anyone involved as a Team Member or Executive Board Member might spend 1000 hours related to WOW. It's an understatement to say that a ton of effort and planning go into making this program special. It's a large-scale project too - there are about 300 to 350 WOW groups each year, consisting of both freshmen and transfer students (about 15 per group) – when you include the leaders, there's approximately 4,000 to 5,000 people directly involved every year.

The reason this project originated is because of my own participation and involvement with the Week of Welcome Program. After spending a great deal of time in the program, it was recognized that there were several areas that could be targeted and improved from an engineering perspective. Being a polytechnic university, there are so many different disciplines that come together to make this college what it is. Many think of engineering as a tool used only in industry – for manufacturing, machinery, and businesses. But this project is a perfect example of how it can be used across multiple disciplines (such as volunteering), and be used effectively.

After much deliberation, it was decided that the emphasis of this project would focus on goal statement number three – the introduction of important issues that new students may encounter. As previously stated, there are approximately 300 to 350 WOW groups each year. Each year there are discussions amongst high-ranking members of the program about what is the most efficient method to enable every group to be educated about these potential encounters. It's typically agreed that the most effective way of accomplishing this is through the creation of a series of awareness presentations that each group is required to attend. And herein lies the problem this project attempts to address. The problem that needs to be solved is creating a schedule that allows every WOW group to see every Awareness Presentation.

Given that problem statement, the word successful needs to be explained in further depth. What does a successful assignment process look like? In short, it means two things – the scheduling process needs to be efficient, and the schedules created need to be effective and satisfactory for each group.

In order to solve the stated problem, a list of objectives was shaped. The following are the goals of the Week of Welcome Group Assignment Problem:

- Study the existing system of WOW group assignment scheduling
- Identify the metrics by which the current and proposed system will be measured (i.e. time or cost)
- Select the areas of the existing system which will be targeted for improvement
- Design a new system for the assignment of WOW groups to the mandatory presentations
- Use identified metrics to show improved performance of the proposed system

To adequately concentrate on these objectives, it is important to form a strategy to help reach them. This requires figuring out what steps need to be taken in order to achieve a solution. While they did not necessary happen in this order, methods were experimented with, tested, and compared in order to find the optimal solution. The following deliverables are the main tasks that will be performed to ensure project completion:

- Use project management skills to establish a timeline
- Conduct research on literature regarding existing systems and how assignment problems have been tackled in the past
- Reach into past records and analyze how this problem has been answered before
- Make use of work study skills to get a feel for the performance of the current system

- Utilize operations research to construct an improved system prototype
- Test and validate new system
- Implement new system into WOW programming
- Summarize all research and findings in a comprehensive report
- Give a final presentation on the project that conveys to all the breadth and range of the project

The end deliverable is simple – a model that the Week of Welcome Program can utilize to help schedule groups into their presentations for the upcoming year, and the future years after that. Just as importantly, it should demonstrate the use of integer programming as an effective tool for consistently solving scheduling problems.

As mentioned, some of the class subjects that will be featured in this report include Process Improvement Fundamentals, Operations Research, Project Management, Systems Engineering, and Engineering Economics.

However, there are details outside of the scope of this project. Based off of data from last year's Week of Welcome, this project will produce a system for assigning all the groups to their awareness presentations. Solving this problem using last year's data will help to provide a model of how to solve the scheduling problems in the upcoming years. Additionally, while there are other things that need to be assigned to groups, such as dining reservations and academic sessions, this system will only focus on producing a system that can optimize the positioning of the awareness programming. However, it will give adequate space in between presentations so there is space to assign the additional events. Furthermore, it will not produce a computer

program. It is beyond the scope of this project to transform the system into a database or a webpage.

In the upcoming sections, details will be given to how that was made possible. This report begins with the background and the literature review, followed by the design section, the methodology, and then the results and the conclusion.

Background

The Week of Welcome is always the week before the start of Fall Quarter and the new school year. While the dates and length have varied, recently it has been coordinated so that the last day of WOW is the last day before classes begin. From the Kick-Off on Day 1 to the Closing Remarks on the last day, there is a ton of work that goes into WOW before, during, and after the Week. While planning for the next WOW starts almost immediately after the previous one ends, the parts relevant to this project start during Spring Quarter. To understand the WOW program, it's important to understand the hierarchy of WOW. Within the WOW infrastructure, there are four main entities – the Executive Board Members, the Team Members, the Orientation Leaders, and the incoming students (also known as WOWies). The Board Members are the top of the structure. Usually consisting of 7 or 8 people, they are in charge of overseeing the entire program. After being selected the year before, they are working nonstop the entire year to help schedule events, handle logistics, communicate with all of the important people at the university, and shape the programming for the entire Week. Board is also in charge of selecting that year's Team. Consisting of around 60 members, Team is like middle management. Team was what I was a part of as well. After applying and being selected in the fall, Team receives their training in the Fall. All of them were orientation leaders at a previous time, if not multiple times. It's up to Team to assist the Board Members, providing support for all of the different jobs Board is tasked with completing. Additionally, Team is held most accountable for training all of the Orientation Leaders in the Spring. During this time, all of the future WOW leaders go through a 10-week training period, spending at least a few hours a week receiving training and preparing for the fall. Each year, anywhere from 800 to 1000 students sign up to be leaders. They have to be trained and educated on everything that is required to be a good leader and capable

representative of Cal Poly. It's up to WOW Team to train and evaluate them. In addition, this is when members of Team are also in committees, working on the content that will be displayed and used during the actual Week.

During the Week, the WOW leaders take their groups all over the campus and the community. Students will get to experience San Luis Obispo in a way that would take them months or even years otherwise. Leaders will sign their groups up for activities such as kayaking, rock climbing, yoga, kickboxing or paintballing. They'll sign them up for events such as seeing a hypnotist, slam poetry, or an improvisational comedy group. They'll reserve dining, both on- and off-campus. But they'll also do more serious things, such as take their groups to academic advising sessions, or lead them on campus tours and highlight important campus resources such as the Health Center. When the Week ends, students are much more equipped to handle the transition to their college experience than they ever would've been otherwise.

And then there's the part of the Week that this project is focusing on – the Awareness Presentations. Throughout the course of the Week, each group has to attend several mandatory presentations, covering subjects such as the consequences of drug and alcohol abuse, the struggles stemming from personal choices, and the appreciation of campus and community diversity. Due to the capacity of the presentation rooms, each of the presentations is shown several times. Some of the presentation times conflict with other presentation times, so the goal is to be able to create a feasible schedule for each group. However, there are a lot more factors to consider to create not just a feasible schedule, but a functional and satisfactory one as well. Each one of those factors just adds another level of complexity to the problem.

Typically this problem has always been tackled manually. Rather than using heuristics or any type of algorithm, it has been solved only by physically inserting each group into all of their

necessary presentations. There is a lot of time and effort each year that goes into making sure each group attends these presentations. They're the bread and butter of the program. They help strike the delicate balance between making sure that incoming freshmen aren't just having fun, but they're being educated on how to become responsible individuals as well.

The effectiveness of this system is ultimately inconclusive, as it is not evaluated on anything except for user feedback. But part of the goal of this project is to be able to find a way to metrically analyze the current system so it can be compared to the new one. Based off how it has been run in the past, it seems like there is a huge opportunity for this system to be reworked into something much more effective and efficient, which would save time for everyone involved.

The common technique for solving this problem is through the utilization of an assignment problem, one of the fundamental optimization tools within the branch of operations research.

The theory behind what this project will do is simple: each of the WOW groups will get assigned to each of their presentations. But actually realizing this and implementing this proves to be a lot tougher than originally imagined. However, this is such a large part of the program, that successfully creating an assignment system is a huge contribution, and a big step forward towards streamlining the entire process. If execution of this system goes successfully, perhaps future goals can involve implementing this on an even broader scale, such as on the items outside of this project's focus.

One of the main complications with solving this problem is that each year the presentations are held in different locations with different capacities, held at different times, and on different days. Not only would it be hard to produce a long-term solution if exact data were used, but this is information that hasn't even been determined yet. A problem can't be solved if variables haven't been defined. So it needs to be noted that when specifics are given, they have been taken from

the most recent data source, the Week of Welcome that took place on the dates of September 14-17, 2012.

Refer to Appendix A to see the timeline for the project and its duration. After that is the literature review, which shows all of the different research that went into tackling this problem and figuring out what the best possible solutions are.

Literature Review

Given the widespread use of planning and scheduling, it was not surprising to see the variety and quantity of articles dedicated to researching various aspects of each subject. However what was surprising was the lack of studies focused specifically on event planning and group scheduling. To the best of the researcher's knowledge, there hasn't been anything involving operations research in this context. While the literature review has returned many results of studies involving assignment and scheduling problems, nothing has been specific to the group-presentation assignment combination that is being addressed. However, many researchers have investigated different aspects of the assignment problem, so it seems logical to instead analyze different scheduling situations and see how they can apply to the problem at hand. A lot of sources may at first appear to be solving different problems and unsuitable for the use of this project, but by thinking creatively and critically, a lot more relevant data can be found.

All articles were systematically searched in the professional databases of ABI/INFORM Complete and Web of Knowledge, and each cited reference was considered relevant for the literature review. The main search terms "operations research", "assignment problem", "goal programming", and "linear programming", were combined with the more specific interests of this project "event planning", "scheduling algorithm", and "group scheduling" in order to find the references stated in the following sections.

The following sections will provide insight into the methods and approaches taken by others to attack certain circumstances, and allowed this project to take a more strongly defined form. Every aspect listed within the literature review was considered by the researcher for his solution,

some of which ended up being implemented. The following is the process which was taken in order to get a firm grasp on the problem at hand.

The literature review began with the intention of getting a better grasp on assignment problems as a whole. Krochmal and Pardalos (2009) completed a study detailing many of the intricacies of the assignment problem, and gave a general overview of what they are and what they are used for. In any assignment problem, it is important to know that "one is looking to find an assignment, or matching between the elements of two (or more) sets, such that the cost of all matched pairs (tuples) is minimized". Krokhmal and Pardalos also noted the numerous ways that an assignment problem can take shape, such as linear, multidimensional, quadratic, and bottleneck. While many assignment problems do focus on cost, this project won't be, hence the objectives will be decided based on different criteria. Additionally, reviewing the different types of assignment problems helped to identify that the group scheduling assignment will take its form as either a linear or multidimensional assignment problem.

One of the most relevant uses of assignment problems for scheduling takes shape in the form of planning the schedule for professional or organized sports leagues and tournaments. Creating a schedule for a sports league has a lot of factors to consider because there are so many different teams, as well as owners, management, and the concerns of television and advertising. Melouk and Keskin created a solution for the National Collegiate Athletic Assocation (NCAA) and their annual basketball tournament. The researchers use integer programming in order to create optimal matchups for the first two rounds of the tournament. Utilizing distance minimization (and the costs associated with travelling) as the objective function permitted them to make use of binary variables to decide whether a team would be assigned to a certain site. The objective function equation effectively determined where each basketball team should be assigned to play

to make the tournament more accessible for student athletes and fans alike, and to reduce the costs associated with staging the tournament. Similar work was done with the Second Division Soccer Leagues in Chile (Duran *et al,* 2012). Similar requirements to the NCAA Tournament were given during formulation as well. They also used integer linear programming to create a regular season playing schedule between all of the Chilean soccer clubs. Their main constraints were travel and economics as well, but also the mandate of having to play each team throughout the course of the season. They also figured out how to use constraints to force matchups such as rivalries that would result in better entertainment and more profit. Another study completed on creating a successful season schedule occurred for the Rocky Mountain Athletic Conference softball league (Saur *et al,*2012). The main characteristic that set them apart from every other scheduling problem is that in order to meet the demands of the schedule in the given time period, they had to play multiple teams on one weekend. This resulted in placing teams into "pods" which would meet in a central location and play every team in the pod. The idea of putting single entities into groups is something worth considering. An assignment problem was also solved in order to create the schedule for the Canadian Football League (Kostuk and Willoughby, 2012). Their schedule was dependent on having contingency plans. With the volatile weather in Canada, and football being played in the winter, it was impossible to know when bad weather was going to strike so they created a model with interchangeable solutions. Each of these articles was crucial in understanding the complexities in large-scale assignment problems.

The idea of assigning certain entities to specific locations is one of the common uses of an assignment problem. Assignment problems are utilized for space utilization and maximizing facility use as well. Jebali *et al* (2006) performed a study that helped to optimize the use of

operating rooms in hospitals. Operating rooms are typically bottlenecks for patient recovery and discharge, and scheduling helped to develop targets for resource-use improvement and cost reductions. The article highlights the limits of resource constraints and the factors that keep from further operations being assigned in one day. Operating hours, the availability of staff, the number of rooms, the available equipment in each room, and the capacity of each room are all constraints in the operating room, but also constraints that can be applied to this project as well. Scheduling of the rooms necessary for the Week of Welcome presentations will be dependent on the availability of the room, the necessary resources to run presentations in the room, and the costs associated with utilizing each room. Another common space-constrained area is the airport. Balakrishnan and Chandran (2010) identified the runway system and operations as the primary bottleneck in any airport terminal. The goal was to take advantage of spacing requirements for planes and improve runway efficiency, but still be able to account for and optimize safety and equity. At the end, their solution utilized the ability to manipulate a first-come-first-serve service system by allowing planes to slides forward or backward a predetermined amount in the service line. They used linear programming to come up with hard constraints that will prove helpful if capacity issues are encountered.

Another industry that heavily utilizes the assignment problem is the truck driving industry. Despite the lack of recognition, many resources and goods are still transported via semi-truck. Goel (2012) was tasked with creating a dynamic programming approach that created a generic model for scheduling routes for truck drivers that also accounts for constraints of the labor laws and hours of service regulations. The dynamic aspect of the program is the ability to account for which route is being utilized and where exactly the rest stops are. In an industry plagued by accidents and poor adherence to labor regulations, creating a schedule that establishes time for

rest is essential. This is an essential tool both for trucking and for this project, because one of the necessary constraints is ensuring that there is ample time in between presentations that groups get scheduled for. Another study was performed by Goel *et al* (2012) where feasible truck driving schedules had to be created in accordance with labor regulations in Australia. The specific regulations in Australia led to the use of a 5-day planning horizon to schedule truck drivers and the time periods when they needed to rest. Considering that the Week of Welcome is also 5 days long, it made sense to consider their schedule.

Assignment problems have further use in employee scheduling in other industries as well. Santos *et al* (2012) took on the task of scheduling teachers to class lectures. Their purpose was to find feasible schedules for each teacher given the hard constraints, and then to create optimal ones given with soft constraints. Hard constraints used involved classes conflicting with each other, assigning only one teacher to a class, and teacher unavailability. Soft constraints included length of lessons, and teaching classes on the same day without gaps in between classes. A lot of these constraints can be viewed as relevant to our problem at hand, because lecture constraints such as having them conflict, and having lectures with gaps between them are the same constraints that will be needed for presentations. Another study by Qi *et al* (2004) details the problem of assigning pilots to training classes. Since pilots need to be trained every time they switch home bases, fleet types, and cockpit seat, there is a great deal of confusion that comes with scheduling them all into classes.

Another aspect of operations research that was considered for this project is the use of goal programming for the formulation. Goal programming incorporates the use of soft constraints, or goals, that can't all be met in a single optimal solution, so instead the goals must be ranked in order of which are most important. Topaloglu and Ozkarahan (2004) formulated a goal

programming model that incorporated employee preferences into their formulation as goals. In addition to employee preferences, they also made use of who had the day off, shift starting times, shift lengths, and break placements to create an optimal schedule. The formulation was consistently altered from day to day when there was a shift in priorities. Depending on what goals needed to be satisfied for a given day, that was what they ranked first in their goal priorities and which constraints became necessary to abide by. Ferland *et al* (2001) were also involved in shift scheduling with the nurses at a hospital. Their decision to use goal programming came from the desire to use the individual goals as penalties. Whenever there was a goal they didn't reach, a penalty was assessed. Aleman *et al* (2007) used goals as a tool to determine search and destroy target priorities for Unmanned Aerial Vehicles (UAVs). In addition to resource constraints such as distance from target, vehicle availability and bomb capacity, the target demands, priorities, and costs to assign UAVs to missions were what determined the boundaries for goal formulation. The benefit of goal programming is the ability to reach near-optimal solutions when an optimal solution is infeasible. However, the final source may have provided the most adept solution to our project.

The work that might be most related to our efforts was performed by Burke *et al* (2012). They examined the shift scheduling of nurses as well. They opted for an iterative approach dependent on hard and soft constraints to satisfy their needs. In order to solve the hard constraints "that must be satisfied in order to have a feasible schedule", they implemented an iterative heuristic that resulted in quick solutions. Solutions came in the form of shift patterns to each nurse. This could translate to the scheduling of presentations by creating "tracks" of specific presentations that each group would follow. But this didn't result in quality schedules, only feasible ones, which have yet to consider the soft constraints. The term used for this is "generation and

allocation", creating the schedules and then randomly assigning them. Allocation is done through binary variables in the objective function, setting them equal to 1 if they are assigned a schedule, and 0 otherwise. The argument can be made that putting a lot of groups (and putting groups into pods) on similar schedules would make presentation schedule assignments much less complex. This leads to the next step. After considering the hard constraints, they focused on the soft constraints where "requirements are desirable but not obligatory" and help to "evaluate the quality of feasible schedules". These include factors such as requests for time off, limited availability, and preferences for shifts. If specific groups needed to deviate from the given schedules for any reason, or had special preferences, this is where they would be implemented. In this paper, improvements are made to this system in order to fit our system. This model is examined closely and proved invaluable as an evaluation tool for strategic decision making.

In short, the conclusion of this literature review is that this senior project is an endeavor never attempted by anyone before. While others have used operations research to assign agents to tasks in different situations, no one has used it to assign groups to presentations. The theory is similar, but the subjects are different. But with these references in mind, the following design was able to reach its full potential.

Design

This section begins with each of the steps that were taken in the engineering design process, that are outlined and summarized, and ends with the solution that was selected for implementation.

The engineering design process is the formula utilized to design a solution to a problem, in this case the Week of Welcome scheduling. The engineering design process is broken down into several stages – defining the problem, finding relevant information, identifying the root causes, generating alternative solutions, doing technical and economic analysis of the possible solutions, selecting the best solution, and then making a recommendation that possibly leads to implementation. The different stages are as follows.

Problem Definition

While appearing simple, defining the problem is one of the most crucial parts of the engineering design process. If the problem is incorrectly defined, it is quite likely that the solution to the problem will be entirely incorrect, and more importantly, will not meet customer requirements. The problem statement has to be free of any bias or reference to the solution or methodology. Referring to the solution in the problem statement implies that the solution was already known beforehand and is thus not a new problem, or that no other alternatives were considered. The process of getting to the solution, or the goals for the project or the tools taken to get there all come later. It's important to start purely with the problem, which is,

"Cal Poly's Orientation Programs needs a time-effective method of assigning incoming-student groups to the Awareness Presentations planned during their Week of Welcome Orientation Program."

Relevant Information

The point of finding relevant information is to know what needs to be considered when working on the initial design. That includes requirements, constraints, goals, and customer specifications that had to be given careful deliberation. When designing the solution, there was at least one of each of those types of information that made important contributions. The first thing that was considered was customer specifications. Because of my own participation in the program and the sentimental value attached, my desire was to create something that they specifically wanted and would be beneficial to them. So before anything else, feedback was obtained from relevant high-ranking members of the program as to how the Week of Welcome went last year, both in planning and execution. In particular, it was important to hear how they felt about the scheduling process and what difficulties they had and what they wish they could've had. Feedback included:

- Quality control and efficiency (make sure no group gets forgotten and that every group sees every presentation)
- Of the several different presentations being given, how many repetitions of each presentation to satisfy the demand (the number of WOW groups and incoming students)
- Difficulty with time conflicts between the presentations and other commitments that had fixed and unchangeable times such as "Day of Service" (community service events); a lot of groups were signed up for it, and had to have their schedules rearranged

It was important to the Week of Welcome Board that the feedback was taken into careful consideration. The first and last pieces of feedback were both utilized in attaining a solution, and the middle piece of feedback was utilized in the formulation constraints.

Physical constraints proved to be an issue in determining where each of the presentations might be placed. Of the presentations that are prepared for each year, there are several venues on the Cal Poly campus where they might be shown. Those sites are Chumash Auditorium in the University Union, Spanos Theatre, the Performing Arts Center, and the Main Gym in the Recreation Center. However, all of those locations have different capacities that have to be considered. The capacities of each place can vary from 500 to 1500. The exact capacities can be seen in Table 1. Also, it was requested that some slack be given between the planned numbers and the actual capacity. The members of the committee who plan all the presentations believe it is necessary to have flexibility in order to things such as selectively add groups to specific presentations or invite special guests to see them as well. Additionally, due to other university scheduled events or construction, occasionally one of the venues won't be open for use, and that needs to be considered as well.

Table 1 The Different Venue Locations and Capacities

Location	Capacity
Chumash	996 seats (chairs or sitting)
Performing Arts Center	1280 seats
Spanos Theatre	500 seats
Rec Center: Main Gym	2022 (1600 bleacher & 422 seats)

In creating a scheduling system, it appeared the biggest concern was all of the uncertainty. Even the venues for next year aren't confirmed yet. For what appeared to start as a straight-forward assignment problem got a lot more complex with all of the unknowns. In the past, when problems were encountered, history and past records were consulted. But if that wasn't an optimal solution to begin with, there are plenty of areas for improvement. This was an important step for this project because after building an initial model off of last year's schedule, it gave way to the main goal of this scheduling system:

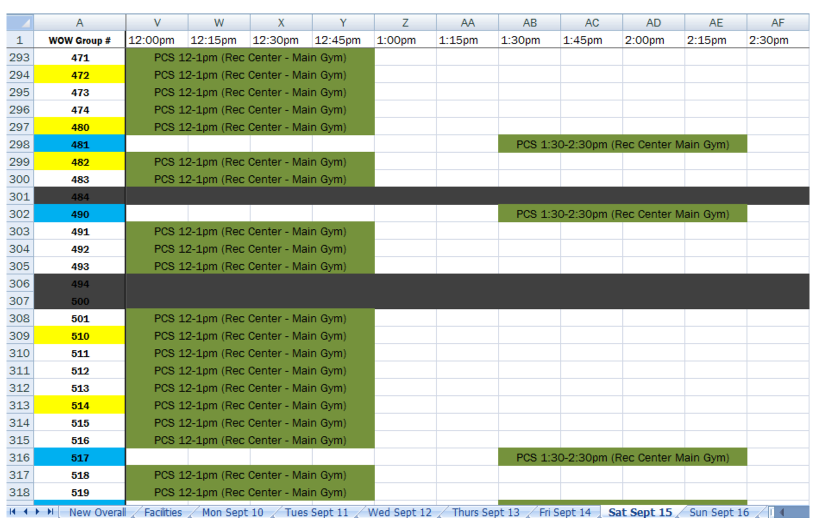
- Because of the given uncertainty and the flexibility that it requires, the goal is to create a general schedule, and be able to "point and shoot" groups into presentations, and

- Create a system that can be reused or recreated by someone besides the designer of the system

What this means is that this system has the potential to be the first step of schedule creation for the entire Week of Welcome, beyond all of the Awareness presentations. Since the presentations are mandatory and a cornerstone of the Week of Welcome Program, it makes sense to start by planning them first and building off of them. The key is to find the right balance of venue utilization, optimal number of repetitions for each presentation, and scheduling each presentation for the most advantageous time. From there, there are several other mandatory events, but are beyond the scope of this project. However, to deal with the uncertainty, it is important to consider the scope of the project. In order to build a system that can be used in a general sense and just slide the presentations into place within a schedule, there would require extensive knowledge of computer programming. While that isn't out of the question for future work, the scope of this senior project involves utilizing the tools within operations research. The best that can be done here is to build a system using last year as the framework and then being able to apply that system to next year when the schedule is decided.

Identifying Root Causes

The main issues that have been encountered year and year again are the quality issues of accidentally forgetting to schedule groups, and the scheduling efficiency



	A	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF
1	WOW Group #	12:00pm	12:15pm	12:30pm	12:45pm	1:00pm	1:15pm	1:30pm	1:45pm	2:00pm	2:15pm	2:30pm
293	471	PCS 12-1pm (Rec Center - Main Gym)										
294	472	PCS 12-1pm (Rec Center - Main Gym)										
295	473	PCS 12-1pm (Rec Center - Main Gym)										
296	474	PCS 12-1pm (Rec Center - Main Gym)										
297	480	PCS 12-1pm (Rec Center - Main Gym)										
298	481							PCS 1:30-2:30pm (Rec Center Main Gym)				
299	482	PCS 12-1pm (Rec Center - Main Gym)										
300	483	PCS 12-1pm (Rec Center - Main Gym)										
301	484											
302	490							PCS 1:30-2:30pm (Rec Center Main Gym)				
303	491	PCS 12-1pm (Rec Center - Main Gym)										
304	492	PCS 12-1pm (Rec Center - Main Gym)										
305	493	PCS 12-1pm (Rec Center - Main Gym)										
306	494											
307	500											
308	501	PCS 12-1pm (Rec Center - Main Gym)										
309	510	PCS 12-1pm (Rec Center - Main Gym)										
310	511	PCS 12-1pm (Rec Center - Main Gym)										
311	512	PCS 12-1pm (Rec Center - Main Gym)										
312	513	PCS 12-1pm (Rec Center - Main Gym)										
313	514	PCS 12-1pm (Rec Center - Main Gym)										
314	515	PCS 12-1pm (Rec Center - Main Gym)										
315	516	PCS 12-1pm (Rec Center - Main Gym)										
316	517							PCS 1:30-2:30pm (Rec Center Main Gym)				
317	518	PCS 12-1pm (Rec Center - Main Gym)										
318	519	PCS 12-1pm (Rec Center - Main Gym)										

New Overall / Facilities / Mon Sept 10 / Tues Sept 11 / Wed Sept 12 / Thurs Sept 13 / Fri Sept 14 / Sat Sept 15 / Sun Sept 16

Figure 1 An example of the current scheduling system

and the time it takes to schedule all of the groups. Identifying the root cause is actually pretty straightforward; there is no formal system in place for creating the schedule and assigning groups to each presentation. Each year, one of the WOW Executive Board Members is given the task of doing this process “by hand”. This just consists of having an Excel Spreadsheet for each day of the Week, with the top row dedicated to times (ranging from 7AM to 1AM), and the first column dedicated to WOW Group number. From there, it is a process of manually assigning presentations to each group. Looking at how it is currently done, it is evident that operations research can be used to design a better solution that will be more helpful in the short term and possibly in the long term as well.

Alternative Solutions

The system went through many transformations to get to where it is now. The original idea wasn’t even close to the final product. But with any task, it’s important to know when to change directions and try something else. Every solution needs to start somewhere. And like with many problems, it was helpful to start with the big picture and then continue to keep narrowing the focus.

First attempt

After conducting the literature review, it seemed like the best path to pursue was utilizing binary programming. There were several initial variables that were critical to the creation of a functioning schedule. Those are who (the WOW groups), what (Awareness Presentations), when (presentation time slots), and where (presentation venues). Initially, it made most sense to use a binary variable with the following specifications:

- $X_{ijkl} = 1$ if Group i was assigned to Presentation j, during Time Slot k, and at Presentation Venue l
- $X_{ijkl} = 0$ otherwise

However, with over 300 groups, several presentations, several time slots for each presentation, and several presentation venues, that put the number of binary variables well into the thousands. Using the Excel Add-On Solver, that was much too large of a problem to handle. So there was a need for extreme simplification. And where that seemed most likely is with the presentations.

Of the four different variables (i, j, k, l), three of them were related to the presentation – the presentation itself, the time it was being offered, and where it was being offered. All of those variables are things that are reasonably beyond the control of this project. The Executive Board won't decide for a few months what presentations are going to be shown, or when and where they will be shown. To put it succinctly, the master schedule has yet to be made. So the question was, how can WOW groups be assigned to a schedule that hasn't been created yet? Once all of the presentations were decided on, and the time and dates were all confirmed, it would be relatively simple to create optimal schedules for each group. But how could a system work where the groups can be just slotted into place where there's no place to slot them?

Second Attempt

To answer that question, two things were required. First, it was necessary to start building a model off of the schedule from last year. Using past data would at least give a good idea of when each of the presentations might be. Second, there needed to be some kind of uniformity with the schedules. If over 300 different groups were going to be on 300 different schedules, it would be complete chaos. But that led to the idea of presentation "tracks". If a bunch of

different tracks were created, each group could just be assigned to a certain track. It wasn't important to know what presentations were on the specific track yet, but groups could be assigned to them prematurely. The only constraint that needed to be initially considered was capacity. And this would be a good way of dealing with the issue of some groups that have time-conflict issues with presentations and Day of Service. They would just be put on a different track. However, with this solution, it would still be complicated to execute. For someone unfamiliar with the system, it would be difficult to understand the multi-step process, where there would first be a binary program where groups get assigned to tracks, and then another process where presentations get assigned to schedules.

And then there was the presentation scheduling itself. There needed to be more than one track for presentations, and it was difficult to figure out how many. Using the capacity of the smallest venue (500 – Spanos Theatre), and the average number of people in each group (15), it was calculated that there should be about 30 groups per schedule. With 30 groups, that would make more or less 450 people on one specific schedule. That was the largest number of people that could be allotted to one schedule without going over the capacity of any presentation venue. Then, taking into account the average number of groups per year (between 300 and 325), and having approximately 30 groups per schedule, that amounted to the need for at least ten or eleven schedules or presentation tracks. But the complexity of finding optimal schedules for eleven tracks was not feasible in the given time frame and also a gross over complication of the problem at hand. To make it simple enough to understand, the goal was to be able to solve it using only one step, to be able to be solved on one Excel spreadsheet. So it was time to learn lessons from this attempt, change directions, and try a different approach.

Third, Fourth, and Fifth Attempts

There was definitely one lesson worth learning from the second attempt – the idea of tracks. Putting a lot of groups on the same presentation schedule was the best way of achieving efficiency and making it simple to figure out what groups were going where and at what times. So instead of putting presentations into schedules and using presentation tracks, individual WOW groups were placed into larger groups which were dubbed pods. Each of those pods would be assigned to the series of presentations they needed to attend. Using past data to create the model, there were 3 presentations, labeled as R&D (Respect and Diversity), DSDR (Drunk Sex or Date Rape, Can You Tell The Difference?), and PCS (Personal Choices and Struggles). There were 3 different showings of R&D, 4 of DSDR, and 7 of PCS (because it was in a smaller auditorium). The whole presentation schedule can be seen in Appendix B. But with only 3 different presentations that needed to be seen, it was much easier to adopt this formulation:

- $X_{ij} = 1$ when presentation i is assigned to pod j
- $X_{ij} = 0$ otherwise
- $\sum_{i=1\ j=1} Xij = 33$
- i = 1, 2, 3,…,14 j = 1, 2, 3

The summation of the objective function is equal to 33 because since there are eleven pods being assigned to three presentations each. Using binary programming, that means the objective function will end up equaling 33. Along with this objective function, there were several constraints. First, each pod had to see each presentation once. That meant three sets of eleven constraints, each set for a different presentation, each constraint for a different pod. Second, each presentation could only support a certain number of pods. This was based off of venue

capacities. This resulted in fourteen constraints, one capacity constraint for each presentation. Finally, there were conflict constraints, so if two presentations coincided, they were unable to be assigned to the same group. There were a total of four time slots where there were time conflicts: Saturday from 10-11AM and 12-1PM, and Sunday from 10-11AM and 12-1PM. Each time conflict needed a separate constraint for each pod dictating that if the pod was assigned to one of these presentations, then they wouldn't be allowed to be assigned to the other. These constraints helped create a feasible solution. The constraints were modeled into the Solver as follows:

Constraint 1 (see each presentation once):

- $X_{1j} + X_{2j} + X_{3j} = 1$(R&D)
- $X_{4j} + X_{5j} + X_{6j} + X_{9j} + X_{10j} + X_{11j} + X_{12j} = 1$ (PCS)
- $X_{7j} + X_{8j} + X_{13j} + X_{14j} = 1$ (DSDR)
- $j = 1, 2, 3, \ldots, 11$

Constraint 2 (capacity):

- $X_{i1} + X_{i2} + X_{i3} + X_{i4} + X_{i5} + X_{i6} + X_{i7} + X_{i8} + X_{i9} + X_{i10} + X_{i11} \leq 4$ (R&D and DSDR)
- $i = 1, 2, 3, 7, 8, 13, 14$
- $X_{i1} + X_{i2} + X_{i3} + X_{i4} + X_{i5} + X_{i6} + X_{i7} + X_{i8} + X_{i9} + X_{i10} + X_{i11} \leq 2$ (PCS)
- $i = 4, 5, 6, 9, 10, 11, 12$

Constraint 3 (time conflict):

- $X_{4j} + X_{7j} \leq 1$
- $X_{5j} + X_{8j} \leq 1$
- $X_{9j} + X_{13j} \leq 1$
- $X_{10j} + X_{14j} \leq 1$

- j = 1, 2, 3,…, 11

After that iteration, two more iterations were performed. The first was expanded to 6 pods (of 50 WOW groups and 750 people), and the second was expanded to 12 pods (of 25 WOW groups and 375 people). Dividing them up even more created the ability for greater flexibility in scheduling. Each of these produced an optimal solution. However, the main flaw with these three iterations was that they were all done with estimates of group sizes which affected the number of pods needed. Also, while the solutions found were technically "optimal" there are still other constraints that could be added to further refine the solution. These are more like minimum viable products than finished solutions.

Sixth Attempt

In the next attempt, I used all of the accurate data from last year as far as the total number of incoming students, the total number of WOW groups, and then had a residence hall breakdown where the number of groups per residence hall was established. The residence halls are crucial because they make the most intrinsic sense for forming pods. The residence hall breakdown can be seen in Table 2. Given the variation in the number of groups per residence hall, it became necessary to group some of them together to help establish pods. The goal was to have as close to 25 groups per pod as possible without exceeding that limit. Exceeding 25 would strain the capacity constraints of the formulation. The easiest way to bunch them was by

Table 2 Number of Groups by Residence Hall

Overall:	Groups:	People: (#grps*15ppl)
Fremont	21	315
Muir	20	300
Santa Lucia	19	285
Sequoia	20	300
Tenaya	21	315
Trinity	22	330
North Mountain	29	435
Cerro Vista	44	660
Sierra Madre	46	690
Yosemite	47	705
OUR	2	30
Transfer	18	270
Band	4	60
TOTAL	313	4695

geographical convenience. Fremont, Muir, Santa Lucia, Sequoia, Tenaya and Trinity are all adjacent to each other and are known as the “Red Bricks”. Also the sum of their combined groups is 123, so it would make sense to bunch all of their WOW groups into 5 pods. Additionally, Cerro Vista and North Mountain are close to each other as well. Combining the two residence halls equals 73 WOW groups which splits into 3 pods nicely. Sierra Madre and Yosemite are both their own separate entities and combining them together wouldn’t do any good, so it’s easy enough to give each of them two pods. And finally, it makes sense to bunch the Transfer Student Groups, the Band Groups and the OUR Groups together because they’re all coming from different places anyway and none of them come from any specific residence halls. All of them can form one pod. Refer to Table 3 for the breakdown. One positive thing to note is this will also work in a general sense from year to year because all of the residence halls are filled to their capacity as it is. There currently isn’t any room for expansion. While all of them were separated into pods cleanly, it presented a problem. This process of division created 13 pods instead of 12 – which would have been fine except for the fact that it exceeded Solver’s solving capacity. With the standard Solver Add-On that comes with Excel, the tool can only handle up to 200 variables and 100 constraints. Because this iteration had 105 constraints, this iteration was left without a solution. There now becomes another need to simplify and reevaluate.

Table 3 Pod Breakdown

Residence Halls	Total Groups	Number of Pods
Red Bricks	123	5
North Mountain/Cerro Vista	73	3
Sierra Madre	46	2
Yosemite	47	2
OUR/Transfer/Band	24	1
Total	313	13

Seventh Attempt/First Solution

The next attempt at a solution had one goal over the previous attempt. Take the 13 pods and narrow it down to fewer. While 12 pods was proven to be feasible, it would create greater

flexibility to go even lower than that. To do that though, it would be necessary to increase pod size. With 313 WOW groups and a pod size of 25 WOW groups, dividing 313 by 25 equals 12.52. That means no matter how the groups were allocated, there could be no fewer than 13 pods. But increasing pod size to 30 WOW groups, would result in 313 divided by 30, equaling 10.43. That means that 11 pods could be achieved. Having 30 groups in a pod also meant that there would be at most 450 people per pod, which was the largest possible amount of people per pod without exceeding the capacity limits of any of the venues (the limiting venue is Spanos Theatre). The one potential drawback to using this method of group division to create pods is that there is no longer a clean separation between residence halls. However, there will be considerable variation in each group's needs, especially when it pertains to whether the group is taking part in Day of Service (the only conflict with the presentations that can't be moved). This just means that creating pods by residence hall can be a way of loosely creating pods, but then WOW groups can be freely substituted between pods in order to cater to specific group needs. With this in mind, the next iteration was performed with 11 pods. This led to a feasible solution that was acceptable to all parties. The schedule details of this solution can be seen in the following section. This is the final formulation of the objective function:

- $X_{ij} = 1$ when presentation i is assigned to pod j
- $X_{ij} = 0$ otherwise
- $\sum_{i=1\ j=1} Xij = 33$
- i = 1, 2, 3,…,14   j = 1, 2, 3, …, 11

This solution is satisfactory in every way but it does highlight three possible unwanted observations or limitations. First, there are two presentations that aren't utilized at all: DSDR on Saturday from 12-1PM (i = 8), and PCS on Sunday from 1:30-2:30PM (i = 11). These can either be removed to save time and resources or they can be used to help mitigate possible risks of overcapacity or to just generally have less people in each presentation. Second, three pods (j = 2,7, 8) have two presentations on the same day, and may ultimately be undesirable because of a lack of balance within the schedules. Finally, it's important to note that there are a total of 91 constraints being used in this iteration. With a total of 11 pods, and a Solver limit of 100 constraints, that implies that if a need for any other constraints came up (such as the desire to have only one presentation per day per pod) and needed to be applied to every pod, Solver wouldn't have the ability to create a feasible solution. However, the first observation can be used to eliminate a presentation and several variables in the process. Therefore it is prudent to find alternative solutions that can remedy some of the concerns. The results of Solution 1 can be seen in the Alternative Solution Analysis section, and the Excel Spreadsheet can be seen in Appendix C.

Second Solution

While a feasible solution has already been found, there is the possibility of creating a different solution that better meets customer needs. At first, there was a thought of increasing pod sizes in order to reduce the number of necessary pod variables. But ultimately 30 WOW groups (and 450 people) was the largest a pod could go without becoming a major inconvenience. A pod any larger than that would eliminate Spanos Theatre as a presentation venue option. But with the current state of fluctuation and uncertainty about the availability of other presentation venues, it was important to have at least the choice to use Spanos Theatre and increase the number of

scheduling options. Additionally, any pod size larger than 500 would make it difficult to make a whole number of pods fit comfortably in any of the other presentation venues as well.

The other clear option for creating a potential improvement was eliminating one of the presentations. The past solutions showed that both one of the PCS presentations and one of the DSDR presentations was superfluous, and that the capacity demand of all of the pods could be met by eliminating a replication of PCS and/or DSDR. So it was decided that presentation(s) would be removed and see how the solution turned out. However, it was important not to do anything too drastic, or anything that might do something damaging to the system. So it was decided to eliminate one of the PCS presentations and go from there. A PCS presentation was selected for several reasons. First, it helped mitigate the risk of potential system sensitivity because there were already a large number of replications (7) of the presentation. One of the PCS presentations would be missed less than a DSDR presentation that has only 4 replications. Second, being in a smaller venue means a lower number of people per presentation. If something went wrong, like one of the presentations needing to be moved elsewhere or the time shifted, it's always easier to move a smaller group of people than a larger one.

To discover how eliminating one of the replications would affect the system, three different approaches were taken. Approach A involved eliminating the PCS Presentation Sunday from 1:30 to 2:30PM ($i = 11$). This was the presentation that went unutilized in the first solution, so it made sense to eliminate this one. Approach B involved eliminate the final PCS presentation, taking place on Sunday from 3 to 4PM ($i = 12$). Getting rid of the last one makes sense because it reduces the duration of the commitments that needed to be made, specifically the availability of the staff needed to run the presentation. Finally, Approach C eliminated the PCS presentation on Sunday from 10-11AM ($i = 9$). This was done for the sole reason because its time slot also

conflicted with another presentation – DSDR (i = 13), and reducing conflict was a sure way of increasing flexibility.

After these three approaches, it became clear that there was little discernible difference in schedule quality when a PCS presentation was eliminated. Each of them still provided a viable solution, however, they all increased the number of schedules that had multiple presentations on the same day. Additionally, Approaches A and B only resulted in one constraint reduction – the capacity constraint for the specific presentation – but Approach C proved more useful because of the additional time conflict constraints that were capable of being eradicated. The results of these three approaches can be found in the Alternative Solution Analysis section. The Excel Spreadsheet for Solution 2c can be seen in Appendix D. Solution 2a and 2b were not included in the Appendix because of redundancy and did not demonstrate anything Solution 2c didn't.

Third Solution

While the previous approaches didn't produce discernibly better schedules, it still accomplished the goal of reducing the amount of constraints, allowing the opportunity to add more constraints that could provide another possible solution. Specifically, it gave the opportunity to add two desired constraints – creating a schedule where each pod only needed to see one presentation each day, and creating constraints that set a minimum of how many pods had to attend each presentation. The set up for the constraints was simple enough. For the one presentation a day constraint, each pod has a separate constraint for Saturday and Sunday (Friday's only presentation is R&D so a constraint is not necessary), where only one of the presentations on each day can be assigned. This results in a total of 22 additional constraints. The formulation is as follows:

- $X_{4j} + X_{5j} + X_{6j} + X_{7j} + X_{8j} = 1$ (for Saturday presentations)
- $X_{9j} + X_{10j} + X_{11j} + X_{12j} + X_{13j} + X_{14j} = 1$ (for Sunday presentations)
- $j = 1, 2, 3, \ldots, 14$

The one presentation a day constraint also had one unforeseen implication that was very useful. Because of these constraints, the time conflict constraints were rendered redundant. With the assurance that only one presentation got scheduled each day, there was no possibility of any of the presentations conflicting with each other. Initially, it was believed that this constraint by itself would be enough. However, the initial iteration showed an obvious unevenness in the amount of pods scheduled to each presentation. For example, four pods would get scheduled to a DSDR presentation, but then only two pods would get scheduled to another DSDR presentation, and only one pod to another one. This lack of balance is unacceptable. And that's what created the need for this next constraint.

For the baseline minimum constraints, the structure is similar to the capacity constraints. Each presentation replication has a constraint. But rather than setting a maximum number of pods that can attend the presentation, it sets a minimum. The minimum was established as two pods for DSDR (since it's in the Recreation Center Main Gym, which can hold up to four pods), and one pod for PCS (because it's in Chumash Auditorium, which can only hold two pods). This results in an extra eleven constraints. Three additional constraints aren't needed for R&D because there currently aren't any redundant replications of that presentation. This is the formulation for this constraint:

- $X_{i1} + X_{i2} + X_{i3} + X_{i4} + X_{i5} + X_{i6} + X_{i7} + X_{i8} + X_{i9} + X_{i10} + X_{i11} \geq 1$
- $i = 4, 5, 6, 9, 10, 11, 12$ (for the PCS presentations)

- $X_{i1} + X_{i2} + X_{i3} + X_{i4} + X_{i5} + X_{i6} + X_{i7} + X_{i8} + X_{i9} + X_{i10} + X_{i11} \geq 2$
- i = 7, 8, 13, 14 (for the DSDR presentations)

The assumption with these constraints was that it would create balance and uniformity in each one of the schedules, so that each group would not only have a quality schedule where they only need to see one presentation a day, but would create equality amongst all of the presentations because the pods would be evenly assigned between all of them. This solution is desirable if there is no desire to eliminate any of the presentations, and to make sure that they are all equally utilized. The results of this solution are also in the following section. The Excel Spreadsheet for Solution 3 can be founded in Appendix E.

While this isn't a list of all the possible solutions, it's more importantly a demonstration of the ability to find a quality solution and an adequate representation of the capabilities that the system has. Specifically it shows that a wide assortment of solutions can be achieved depending on user specifications. For the purposes of this project, it's not necessary to find all of the different iterations, but rather exhibit that the system has the ability to find them. However, that isn't to say that an optimal solution won't be chosen. It will be based on criteria that were important last year and will be important next year.

Alternative Solution Analysis

The following five tables are the results from the first solution, the three different approaches to the second solution, and the two iterations of the third solution. The pod numbers that are in red signify schedules that might be potentially unsatisfying. Each of these schedule results was based on several different qualities such as scheduling flexibility, schedule balance and spacing,

and schedule quality. The finer aspects of each of those qualities and how they were used to evaluate the different options will be detailed in the Methodology and Results sections.

Table 4 Presentation Schedule by Pod – Solution 1

Pod	Schedule
1	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
2	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sat 1:30-2:30PM
3	R&D Fri 5-6PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
4	R&D Fri 3:15-4:15PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
5	R&D Fri 3:15-4:15PM, PCS Sat 12-1PM, DSDR Sun 10-11AM
6	R&D Fri 3:15-4:15PM, PCS Sat 12-1PM, DSDR Sun 10-11AM
7	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, Sun PCS 3-4PM,
8	R&D Fri 2-3PM, Sun DSDR 12-1PM, Sun PCS 3-4PM
9	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 12-1PM
10	R&D Fri 2-3PM, PCS Sat 10-11AM, Sun DSDR 12-1PM
11	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 12-1PM

Table 5 Presentation Schedule by Pod - Solution 2a

Pod	Schedule
1	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sat 1:30-2:30PM,
2	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
3	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
4	R&D Fri 5-6PM, PCS Sat 12-1PM, DSDR Sun 10-11AM
5	R&D Fri 5-6PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
6	R&D Fri 5-6PM, PCS Sat 10-11AM, DSDR Sun 12-1PM
7	R&D Fri 2-3PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
8	R&D Fri 3:15-4:15PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
9	R&D Fri 3:15-4:15PM, DSDR Sun 12-1PM, PCS Sun 1:30-2:30PM
10	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 1:30-2:30PM
11	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 12-1PM

Table 6 Presentation Schedule by Pod - Solution 2b

Pod	Schedule
1	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sat 1:30-2:30PM,
2	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
3	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
4	R&D Fri 5-6PM, PCS Sat 12-1PM, DSDR Sun 10-11AM
5	R&D Fri 5-6PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
6	R&D Fri 5-6PM, PCS Sat 10-11AM, DSDR Sun 12-1PM
7	R&D Fri 2-3PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
8	R&D Fri 3:15-4:15PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
9	R&D Fri 3:15-4:15PM, DSDR Sun 12-1PM, PCS Sun 3-4PM
10	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 3-4PM
11	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 12-1PM

Table 7 Presentation Schedule by Pod - Solution 2c

Pod	Schedule
1	R&D Fri 2-3PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
2	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 1:30-2:30PM
3	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 12-1PM
4	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sat 12-1PM
5	R&D Fri 5-6PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
6	R&D Fri 5-6PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
7	R&D Fri 2-3PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
8	R&D Fri 3:15-4:15PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
9	R&D Fri 3:15-4:15PM, DSDR Sun 12-1PM, PCS Sun 3-4PM
10	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 3-4PM
11	R&D Fri 3:15-4:15PM, DSDR Sun 10-11AM, PCS Sun 12-1PM

Table 8 Presentation Schedule by Pod - Solution 3a

Pod	Schedule
1	R&D Fri 2-3PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
2	R&D Fri 2-3PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
3	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 12-1PM
4	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sun 12-1PM
5	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
6	R&D Fri 5-6PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
7	R&D Fri 2-3PM, PCS Sat 10-11AM, DSDR Sun 12-1PM
8	R&D Fri 3:15-4:15PM, PCS Sat 10-11 AM, DSDR Sun 10-11AM
9	R&D Fri 3:15-4:15PM, PCS Sat 1:30-2:30PM, DSDR Sun 10-11AM
10	R&D Fri 3:15-4:15PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
11	R&D Fri 3:15-4:15PM, DSDR Sat 12-1PM, PCS Sun 3-4PM

Table 9 Presentation Schedule by Pod - Solution 3b

Pod	Schedule
1	R&D Fri 5-6PM, PCS Sat 12-1PM, DSDR Sun 12-1PM
2	R&D Fri 5-6PM, PCS Sat 1:30-2:30PM, DSDR Sun 10-11AM
3	R&D Fri 5-6PM, PCS Sat 12-1PM, DSDR Sun 10-11AM
4	R&D Fri 3:15-4:15PM, PCS Sat 1:30-2:30PM, DSDR Sun 12-1PM
5	R&D Fri 3:15-4:15PM, DSDR Sat 10-11AM, PCS Sun 1:30-2:30PM
6	R&D Fri 3:15-4:15PM, PCS Sat 10-11AM, DSDR Sun 10-11AM
7	R&D Fri 3:15-4:15PM, DSDR Sat 10-11AM, PCS Sun 10-11AM
8	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 1:30-2:30PM
9	R&D Fri 2-3PM, DSDR Sat 12-1PM, PCS Sun 10-11AM
10	R&D Fri 2-3PM, DSDR Sat 12-1PM, PCS Sun 3-4PM
11	R&D Fri 2-3PM, DSDR Sat 10-11AM, PCS Sun 12-1PM

Methodology

This section consists of the means used to evaluate the different potential design solutions of the assignment problem application. It begins with establishing what metrics are important to validate the system, and finishes with what method was used to evaluate and select the final design.

When describing the success of any system, there is a trio of requirements that must be met – schedule, customer, and budget. As noted earlier, the Gantt Chart for the senior project is in Appendix A. For this project, the main importance of the schedule is mainly that the project was completed in the time frame of the senior project class. That isn't really a concern. The customer requirements will be covered next. But what's interesting about this project is the budget. Since WOW is a volunteer-run program, there aren't a lot of financial costs associated with the work done by members of the program. Being a part of the program is considered a privilege, a way of giving back, and just a lot of fun in general. While the Executive Board Members do get paid wages during the summer before the Week, the amount they are compensated is rather marginal and possibly negligible. Regardless, these costs will be noted in the analysis.

However, time savings is where true system improvement will be determined. Since costs won't tell the true story, the decrease in the amount of time it takes to utilize the system will justify whether the solution is a substantial upgrade. Time savings will be realized in two ways – Executive Board Member work hours and improvements in WOW group schedules. It's easy to understand why Board Member work hours are important. One or two of the Board Members is tasked with creating the Awareness Presentation schedule and assigning groups to it. It typically takes a couple of 8-hour days, and if that amount of time can be cut down, their time can be used

more purposefully elsewhere instead of on the design of the system. Improvements in WOW group schedules were a little more abstract. It might be easier to understand if it's considered within the realm of user satisfaction. So the question for users: what is an improvement in each group's schedule? There are varying opinions, but that was solved by surveying the people whose opinion matter most – the Executive Board Members.

Out of the questions they were asked, two are most relevant for this justification. The first question, "As a Group Leader, would you rather have all of the presentations on different days, or have as many as possible bunched together?" The second question, "If presentations were on the same day, what is the ideal amount of time to use as spacing between the presentations?" The answer to the first question was decidedly for presentations on different days. The explanation being that a lot of these presentations have pretty deep emotional messages that can affect people in large variety of ways; it's important that incoming students are given the opportunity to let all the material sink in, and enough time to process everything that was presented. Putting presentations on the same day might lead to an overload of emotions, and might take away from the profoundness of the programming. Additionally, from a logistical standpoint, having multiple presentations on one day also makes it difficult to schedule other activities in between presentations (because of the need to schedule in time for rest, eating, travel, etc.), and also may not leave as much time to schedule activities for after. However, if presentations were on the same day, it was agreed that they either need to be at least three hours apart (to give enough time for it to sink in, as well as possibly include other activities in between), or to have them less than an hour apart (so they can be seen essentially back-to-back, so there's more opportunity to take advantage of the remainder of the day. It should be noted

though that having at least three hours of down time was more preferred than only having less than one hour of break in between.

So to make this a tangible justification, it was necessary to go back and look at the data from last year. There were two important things that were looked for in the schedule of each group. The first being, how many WOW groups had schedules where they attended presentations on the same day. The second being, of those WOW groups, how many of them had breaks of one to three hours between those presentations. Any instance of either of those is an opportunity for an improvement.

Time was only one of the considerations for validating the system. The other considerations came in the form of customer requirements. These other requirements were highly valued in selecting one of the solutions for this project. One of the most important was schedule flexibility. That was the reason for finding an assortment of different schedules that might be satisfactory. Having all of these different types of solutions is a demonstration of the fact that creating a quality schedule can be done in several different ways. With only several strict requirements and an ample amount of presentation offerings, the determining factor will be known when they decide what the most pressing needs are for next year. In order to be ready for the future, the most important criterion is the ability of the schedule to cater to whatever the needs end up being. What that ends up meaning is that the solver formulation needs to have the right combination of the minimum numbers of presentations, pods, and constraints. The more space there is between the solution and the Solver capacity limits, the more opportunity there is to add additional constraints. Refer to Appendix F to see how each of the different solutions compares in each of the key statistical categories.

Other than flexibility, there were several other criteria that were used to evaluate the potential solutions. First, balance in each of the pods' schedules. In the organization of each pod's schedules, having adequate time and spacing between presentations is a necessity for being able to plan activities into a WOW Group's schedule. Another concern is the likelihood of committing errors in the scheduling process. The solution has to help reduce the tendency of making mistakes. Next, it's important for the presentations in each solution to be relatively insensitive to being overcapacity. The main point of this criterion is that if the amount of presentations being offered is minimized, it'll strain presentation capacity and present the chance that venues might end up overfilled. After that, ease of understanding is another important quality. The person in charge of scheduling may not have an engineering background, so it is important that he/she is capable of learning the system. Finally, it's important to efficiently utilize the program's time, energy, and resources. There are a lot of things to be done during the Week, so if there is staff that isn't needed to run a presentation, they will be much better utilized elsewhere.

To help compare and analyze each of these attributes, a Pugh Chart was utilized. A Pugh Chart is essentially a ranking tool, comparing potential solutions against the current norm; by assigning a weight to each of the involved attributes and a +/- ranking system, each of the different solutions is assigned a numerical score for each attribute and then the total scores determine what the best alternative is. The conclusion of the Pugh chart will be seen in the following section.

The Pugh chart isn't essential for establishing that using binary programming is better than doing it by hand, but it is helpful in determining which of the various solutions works best for the model of last year's schedule.

Analysis Results and Recommendation

This section consists of the outcomes of the methods used to analyze the current and proposed systems. It starts with showing the statistics that were obtained from doing time, system and solution analysis, and it finishes with the final recommendation to be used as the model for all future work.

Before analyzing any of the solutions individually, it's important to analyze them as a whole against the current system. Where they all excel is in the time it takes to formulate the schedule. From beginning to end, the current system takes about 15 hours. That includes creating the schedule within Excel (the top row indicating time in hourly slots and the first column listing WOW Group numbers, creating multiple spreadsheets for each different day), assigning all of the groups to the presentations they needed to attend, checking it for accuracy, and fixing any errors. The last two steps take the longest, because multiple people need to proofread it and catch any errors that others have missed. However, this isn't a completely fool-proof method. After doing the analysis of the current system for this project, several groups were found to be missing presentation assignments.

While ultimately one of the proposed solutions may be better than the others, it is important to note that each of these represents a marked improvement in formulation speed than the current system. The entire process – setting up the variables, creating the objective function and constraints, running the Solver, tinkering with the solution – took approximately 10 hours. But after creating the initial solution and understanding how the solution was going to take shape, each successive attempt took less than an hour each. The total amount of time to create all of the different variations on the system took about 20 hours. While that may seem like more time than it took than the current system, it should again be noted that this system will be used in the

future. When this system is used for its actual purpose, having the model in place will ensure that there is less of a learning curve and that the formulation process will be completed much quicker than it took initially. A better reflection is the average time it took to create each of the solution attempts. Taking the twenty hours of formulations and the five solutions that were created, formulating a future system should only take four or five hours. Since the Board Members get paid $12 an hour during the summer, and the amount of time savings is about 10 hours, the final cost savings are in the range of $120. While that isn't a typically large number, for a volunteer organization such as WOW, that still has practical significance.

When it comes to user satisfaction, the solution that incorporated the one presentation a day constraint proved to be quite valuable. Using that solution, there is a much more substantial improvement that can be made on the current system. The schedule from last year was examined to see how many groups had presentations where there was more than 1 hour and less than 3 hours between their presentations. After looking at all of the schedules, there were 120 WOW groups with inadequate presentation scheduling. Since they had anywhere from 1 to 3 hours of space in between their presentations, that is an average of about 2 hours of space between presentations that isn't enough time to both process the information and plan another activity. That's a total of 240 hours that WOW groups will be able to utilize more productively. Since WOW groups only have a few days to plan activities, these savings are crucial. Every hour is important to maximizing the productivity of each group as well as the amount of recreational activities they get to partake in. Both the time savings in formulation and the increase in user satisfaction with their schedules demonstrate that the proposed system(s) are an upgrade over the current system.

However, for the purposes of this senior project, it's important to make a recommendation for the problem at hand. That problem is the scheduling matter from last year's Week of Welcome. To solve that, the criteria listed in the previous section were inserted into a Pugh Chart, as can be seen in Appendix G.

The first thing that was established was the weights of each of the criteria. Based off of customer requirements and the goals of this project, the most important qualities were flexibility for future use and ease of understanding for someone not familiar with the system. That's why they were given a weight of three. This ensures that they are taken most seriously in the decision process. When it comes to flexibility, Solutions 2 and 3 are the most capable of being manipulated in the future. Solution 2 has a presentation removed from the formulation which resulted in fewer objective function variables, but Solution 3 has the fewest constraints. Either way, both excel in creating maneuverability. However, all three solutions are unsuccessful in being easy to understand by someone unfamiliar with the system. The reason for this is the nature of operations research and integer programming. They are concepts that aren't common knowledge and are therefore difficult to use without background knowledge.

The qualities that were viewed as secondary in importance were the presentation schedule organization and the likelihood of committing errors. These were assigned a weight of two. Presentation schedule organization is directly related to having balance in each of the pod schedules, whether that's having one presentation a day, or adequate spacing between them if they are on the same day. This is where Solution 3 excels as it has constraints that were designed specifically for this requirement. When it comes to committing errors, every solution is an improvement in the area of quality control. While there is no direct fail-safe to keep groups from being passed over or forgotten, having them grouped into pods creates a higher level of

accountability. So instead of just checking that every WOW group is accounted for, it's also a matter of making sure that every pod and every group within that pod is accounted for.

The qualities that were deemed least important were presentation venues having potential of being overcapacity and the amount of time, energy and resources needed to run the presentations. Each year, the number of presentations that are offered exceeds how many are actually needed. That's why the majority of these solutions ended up not utilizing one of the presentations. But the reason this is done is so that there is no potential for being over capacity. Additionally when setting the initial targets for people in each of the venues, a buffer was created between the target and the actual size so that overcapacity was highly unlikely. However, there is some fluctuation in the size of WOW groups and the possibility of additional guests, so overcapacity is a remote possibility. Since Solution 2 is the only scheduling system that eliminated one of the presentations, it is the only one with any danger of having overfilled presentations. On the other hand, Solution 3 utilizes every available presentation and also uses the baseline constraint to create pod equity between presentations, so there is almost no chance of a venue going beyond the maximum. However, since Solution 3 utilizes every presentation, it requires more effort to produce, since both Solution 1 and 2 have multiple presentations that aren't being utilized and thus don't have to use up any resources. The final results of the Pugh Chart show that Solution 3 is the most well-rounded option and is the best recommendation.

Selection

The Pugh Chart was very influential in making a final decision, but was not the ultimate decider. Regardless, it is still essential to take into account every factor and consider the aspects of every

solution. However, Solution 3 is still regarded as the best scheduling solution for last year's Week of Welcome Programming.

To start, Solution 3 provides the best balance out of any of the schedules. Using the one presentation a day constraint as well as the presentation baseline minimum constraint, it ensures that each pod has a similar schedule and that each presentation is properly utilized. Solution 3 excels in schedule organization and maintaining a buffer between presentation attendance and capacity. But most importantly, Solution 3 has excellent flexibility to future exploitation. The extra variable and constraint flexibility allows for complete control of the schedule. Some possibilities are front-loading the presentation schedule and putting them on the first couple days, creating a schedule where some groups only go to morning presentations and some go to evening presentations, a solution that doesn't utilize one of the presentation venues, there's even the possibility to make the pods smaller so that they are better representations of each of the residence halls. But whatever the need, Solution 3 can provide that option.

Conclusion

Despite not being the solution to a new problem, this project has succeeded in not just solving an old problem but also in showing the necessary methods for solving a future problem. Before this project was started, the aspiration was to create an assignment process for the Week of Welcome Program – a system that created schedules for every WOW group. But in the problem definition stage of the project, it was recognized that that wasn't possible. The final schedule for the presentations isn't determined until a few months before the Week of Welcome begins. Unfortunately, that is also a few months after the conclusion of this project.

There were a couple ways this problem could be solved. The first of which is through the use of a computer program. That would've been the most direct way of achieving what the customers have defined as a "point and shoot" program, where the presentation times can just be inserted into the program and then have the program spit out the ideal solution. However, that sort of programming is more suited to a software or computer engineer, and would be near impossible without that expertise. Regardless, even without a computer program, or without the certainty of next year's presentations, it has now been proved that a successful scheduling system can be created with the given information.

Additionally, even though there weren't monumental savings as far as the overall time and cost savings, there was still a noticeable increase in the customer satisfaction of all of the WOW Groups involved. And as mentioned before, since WOW is a volunteer organization not running off of profit, any improvement at all is extremely beneficial.

And it is those benefits that lead to the most important point. If these benefits can be realized in this one aspect of Cal Poly Orientation Programs, it shows the potential for them to be

discovered everywhere else as well. Future actions will begin with next year's assignment problem. Once the presentation schedule is established, this system will be implemented. If the system proves successful, steps will be taken to see that operations research and linear programming will be considered for future problems as well.

References

Aleman, Rafael, Xinhui Zhang and Raymond Hill. "Application of Multiperiod Generalized Assignment Problem to UAV Planning." *IIE Annual Conference. Proceedings* (2007): 269-274.

Balakrishnan, Hamsa and Bala G Chandran. "Algorithms for Scheduling Runway Operations Under Constrained Position Shifting." *Operations Research* 58.6 (2010): 1650-1665.

Burke, Edmun K, Jingpeng Li and Rong Qu. "A Pareto-based Search Methodology for Multi-Objective Nurse Scheduling." *Annals of Operation Research* 196.1 (2012): 91-109.

Duran, Guillermo, Mario Guajardo and Rodrigo Wolf-Yadlin. "Operations Research Techniques for Scheduling Chile's Second Division Soccer League." *Interfaces* 42.3 (2012): 273-286.

Ferland, Jacques A, et al. "Generalized Assignment Type Goal Programming Problem: Application to Nurse Scheduling." *Journal of Heuristics* 7.4 (2001): 391-413.

Goel, Asvin. "The Minimum Duration Truck Driver Scheduling Problem." *SSRN Working Paper Series* (2012).

Goel, Asvin, Claudia Archetti and Martin Savelsbergh. "Truck Driver Scheduling in Australia." *Computers & Operations Research* 39.5 (2012): 1122-1132.

Jebali, A, ABH Alouane and P Ladet. "Operating Rooms Scheduling." *International Journal of Production Economics* 99.1,2 (2006): 52-62.

Kostuk, Kent J and Keith A Willoughby. "A Decision Support System for Scheduling the Canadian Football League." *Interfaces* 42.3 (2012): 286-297.

Krokhmal, Pavlo A and Panos, M. Pardalos. "Random Assignment Problems." *European Journal of Operation Research* 194.1 (2009): 1-17.

Masoud, Mahmoud, Erhan Kozan and Geoff Kent. "A Job-Shop Scheduling Approach for Optimizing Sugarcane Rail Operations." *Flexible Services and Manufacturing Journal* 23.2 (2011): 181-206.

Melouk, S H and B B Keskin. "Team Assignments and Scheduling for the NCAA Basketball Tournament." *The Journal of the Operational Research Society* 63.5 (2012): 620-632.

Qi, Xiangtong, Jonathan F Bard and Gang Yu. "Class Scheduling for Pilot Training." *Operations Research* 52.1 (2004): 148-162.

Santos, Haroldo Gambini, et al. "Strong Bounds with Cut and Column Generation for Class-Teacher Timetabling." *Annals of Operation Research* 194.1 (2012): 399-412.

Saur, Marjorie Cone, et al. "Scheduling Softball Series in the Rocky Mountain Athletic Conference." *Interfaces* 42.3 (2012): 296-310.

Topaloglu, Seyda and Irem Ozkarahan. "An Implicit Goal Programming Model for the Tour Scheduling Program Considering the Employee Work Preferences." *Annals of Operation Research* 128.1-4 (2004): 135-158.

Appendices

Appendix A: Gantt Chart for the Entire Project

	Quarter 1											
Dates (Week of)	17-Sep	24-Sep	1-Oct	8-Oct	15-Oct	22-Oct	29-Oct	5-Nov	12-Nov	19-Nov	26-Nov	3-Dec
Week	1	2	3	4	5	6	7	8	9	10	11	12 (Finals)
Project Selection/Confirmation	■	■	■	■								
Project Proposal				■	■							
Literature Review					■	■	■					
Project Status Report #1						■						
Final Draft - Introduction, Background, Lit Review							■	■	■			
Project Work - Evaluate Current System										■	■	■
Project Work - Design New System												
Project Status Report #2												
Project Report												
Presentation												
Celebration												

	Winter Break				Quarter 2										
Dates (Week of)	10-Dec	17-Dec	24-Dec	31-Dec	7-Jan	14-Jan	21-Jan	28-Jan	4-Feb	11-Feb	18-Feb	25-Feb	4-Mar	11-Mar	18-Mar
Week	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27 (Finals)
Project Selection/Confirmation															
Project Proposal															
Literature Review															
Project Status Report #1															
Final Draft - Introduction, Background, Lit Review															
Project Work - Evaluate Current System	■	■	■	■	■	■									
Project Work - Design New System					■	■	■	■	■	■					
Project Status Report #2						■									
Project Report									■	■	■	■	■	■	
Presentation									■	■	■				
Celebration															■

Appendix B: Presentation Schedule for Week of Welcome September 13-17, 2012

Name	Location	Date	Time Start	Time End	i
Respect & Diversity	Rec Center: Main Gym	Fri 9/14/12	2:00PM	3:00PM	1
Respect & Diversity	Rec Center: Main Gym	Fri 9/14/12	3:15PM	4:15PM	2
Respect & Diversity	Rec Center: Main Gym	Fri 9/14/12	5:00PM	6:00PM	3
Personal Choices and Struggles	Chumash	Sat 9/15/12	10:00AM	11:00AM	4
Personal Choices and Struggles	Chumash	Sat 9/15/12	12:00PM	1:00PM	5
Personal Choices and Struggles	Chumash	Sat 9/15/12	1:30PM	2:30PM	6
Drunk Sex or Date Rape	Rec Center: Main Gym	Sat 9/15/12	10:00AM	11:00AM	7
Drunk Sex or Date Rape	Rec Center: Main Gym	Sat 9/15/12	12:00PM	1:00PM	8
Personal Choices and Struggles	Chumash	Sun 9/16/12	10:00AM	11:00AM	9
Personal Choices and Struggles	Chumash	Sun 9/16/12	12:00PM	1:00PM	10
Personal Choices and Struggles	Chumash	Sun 9/16/12	1:30PM	2:30PM	11
Personal Choices and Struggles	Chumash	Sun 9/16/12	3:00PM	4:00PM	12
Drunk Sex or Date Rape	Rec Center: Main Gym	Sun 9/16/12	10:00AM	11:00AM	13
Drunk Sex or Date Rape	Rec Center: Main Gym	Sun 9/16/12	12:00PM	1:00PM	14

Appendix C: Excel Spreadsheet for Solution 1

Variables		X1,1	X1,2	X1,3	X1,4	X1,5	X1,6	X1,7	X1,8	X1,9	X1,10	X1,11	X2,1	X2,2	X2,3	X2,4	X2,5	X2,6	X2,7	X2,8	X2,9	X2,10	X2,11	X3,1	X3,2	X3,3	X3,4	X3,5	X3,6	X3,7	X3,8	X3,9	X3,10	X3,11
Objective Function Coefficients		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Variable Place Holders		0	0	0	0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
Constraints																																		
Constraint 1:	Group 1	1											1											1										
Each pod will see R&D	Group 2		1											1											1									
(i=1,2,3) once	Group 3			1											1											1								
	Group 4				1											1											1							
	Group 5					1											1											1						
	Group 6						1											1											1					
	Group 7							1											1											1				
	Group 8								1											1											1			
	Group 9									1											1											1		
	Group 10										1											1											1	
	Group 11											1											1											1
Constraint 2:	Group 1																																	
Each pod will see PCS	Group 2																																	
(i=4,5,6,9,10,11,12) once	Group 3																																	
	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	
Constraint 3:	Group 1																																	
Each pod will see DSDR	Group 2																																	
(i=7,8,13,14) once	Group 3																																	
	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	
Constraint 4: Capacity	R&D (i=1)	1	1	1	1	1	1	1	1	1	1	1																						
R&D can have 4 pods	R&D(i=2)												1	1	1	1	1	1	1	1	1	1	1											
per presentation (Rec)	R&D(i=3)																							1	1	1	1	1	1	1	1	1	1	1
Constraint 5: Capacity	PCS (i=4)																																	
PCS can have 2 pods	PCS (i=5)																																	
per presentation	PCS (i=6)																																	
(due to Chumash capacity)	PCS (i=9)																																	
	PCS (i=10)																																	
	PCS (i=11)																																	
	PCS (i=12)																																	
Constraint 6: Capacity	DSDR(i=7)																																	
DSDR can have 4 pods	DSDR(i=8)																																	
per presentation	DSDR(i=13)																																	
(due to Rec Ctr capacity)	DSDR(i=14)																																	
Constraint 7: Conflict	Group 1																																	
If any Group is scheduled for	Group 2																																	
Pres. i=4, they can't be in	Group 3																																	
Pres. i=7, and vice versa	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	
Constraint 8: Conflict	Group 1																																	
If any Group is scheduled for	Group 2																																	
Pres. i=5, they can't be in	Group 3																																	
Pres. i=8, and vice versa	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	
Constraint 9: Conflict	Group 1																																	
If any Group is scheduled for	Group 2																																	
Pres. i=9, they can't be in	Group 3																																	
Pres. i=13, and vice versa	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	
Constraint 10: Conflict	Group 1																																	
If any Group is scheduled for	Group 2																																	
Pres. i=10, they can't be in	Group 3																																	
Pres. i=14, and vice versa	Group 4																																	
	Group 5																																	
	Group 6																																	
	Group 7																																	
	Group 8																																	
	Group 9																																	
	Group 10																																	
	Group 11																																	

X4,1	X4,2	X4,3	X4,4	X4,5	X4,6	X4,7	X4,8	X4,9	X4,10	X4,11	X5,1	X5,2	X5,3	X5,4	X5,5	X5,6	X5,7	X5,8	X5,9	X5,10	X5,11	X6,1	X6,2	X6,3	X6,4	X6,5	X6,6	X6,7	X6,8	X6,9	X6,10	X6,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1																					
												1																				
													1																			
														1																		
															1																	
																1																
																	1															
																		1														
																			1													
																				1												
																					1											

X7,1	X7,2	X7,3	X7,4	X7,5	X7,6	X7,7	X7,8	X7,9	X7,10	X7,11	X8,1	X8,2	X8,3	X8,4	X8,5	X8,6	X8,7	X8,8	X8,9	X8,10	X8,11	X9,1	X9,2	X9,3	X9,4	X9,5	X9,6	X9,7	X9,8	X9,9	X9,10	X9,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
																						1										
																							1									
																								1								
																									1							
																										1						
																											1					
																												1				
																													1			
																														1		
																															1	
																																1
1											1																					
	1											1																				
		1											1																			
			1											1																		
				1											1																	
					1											1																
						1											1															
							1											1														
								1											1													
									1											1												
										1											1											
																						1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1																					
												1																				
													1																			
														1																		
															1																	
																1																
																	1															
																		1														
																			1													
																				1												
																					1											
																						1										
																							1									
																								1								
																									1							
																										1						
																											1					
																												1				
																													1			
																														1		
																															1	
																																1

X10,1	X10,2	X10,3	X10,4	X10,5	X10,6	X10,7	X10,8	X10,9	X10,10	X10,11	X11,1	X11,2	X11,3	X11,4	X11,5	X11,6	X11,7	X11,8	X11,9	X11,10	X11,11	X12,1	X12,2	X12,3	X12,4	X12,5	X12,6	X12,7	X12,8	X12,9	X12,10	X12,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						

X13,1	X13,2	X13,3	X13,4	X13,5	X13,6	X13,7	X13,8	X13,9	X13,10	X13,11	X14,1	X14,2	X14,3	X14,4	X14,5	X14,6	X14,7	X14,8	X14,9	X14,10	X14,11			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
0	0	1	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	33		
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
1											1											1	=	1
	1											1										1	=	1
		1											1									1	=	1
			1											1								1	=	1
				1											1							1	=	1
					1											1						1	=	1
						1											1					1	=	1
							1											1				1	=	1
								1											1			1	=	1
									1											1		1	=	1
										1											1	1	=	1
																						4	<=	4
																						4	<=	4
																						3	<=	4
																						2	<=	2
																						2	<=	2
																						2	<=	2
																						1	<=	2
																						2	<=	2
																						0	<=	2
																						2	<=	2
																						4	<=	4
																						0	<=	4
1	1	1	1	1	1	1	1	1	1	1												4	<=	4
											1	1	1	1	1	1	1	1	1	1	1	3	<=	4
																						1	<=	1
																						1	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						1	<=	1
																						1	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						1	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
1																						1	<=	1
	1																					0	<=	1
		1																				1	<=	1
			1																			0	<=	1
				1																		1	<=	1
					1																	1	<=	1
						1																1	<=	1
							1															0	<=	1
								1														0	<=	1
									1													0	<=	1
										1												0	<=	1
											1											0	<=	1
												1										0	<=	1
													1									0	<=	1
														1								1	<=	1
															1							0	<=	1
																1						0	<=	1
																	1					0	<=	1
																		1				1	<=	1
																			1			1	<=	1
																				1		1	<=	1
																					1	1	<=	1

Appendix D: Excel Spreadsheet for Solution 2c

Variables		X1,1	X1,2	X1,3	X1,4	X1,5	X1,6	X1,7	X1,8	X1,9	X1,10	X1,11	X2,1	X2,2	X2,3	X2,4	X2,5	X2,6	X2,7	X2,8	X2,9	X2,10	X2,11
Objective Function Coefficients		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Variable Place Holders		1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
Constraints																							
Constraint 1:	Group 1	1											1										
Each pod will see R&D	Group 2		1											1									
(i=1,2,3) once	Group 3			1											1								
	Group 4				1											1							
	Group 5					1											1						
	Group 6						1											1					
	Group 7							1											1				
	Group 8								1											1			
	Group 9									1											1		
	Group 10										1											1	
	Group 11											1											1
Constraint 2:	Group 1																						
Each pod will see PCS	Group 2																						
(i=4,5,6,9,10,11,12) once	Group 3																						
	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 3:	Group 1																						
Each pod will see DSDR	Group 2																						
(i=7,8,13,14) once	Group 3																						
	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 4: Capacity	R&D (i=1)	1	1	1	1	1	1	1	1	1	1	1											
R&D can have 4 pods	R&D(i=2)												1	1	1	1	1	1	1	1	1	1	1
per presentation (Rec)	R&D(i=3)																						
Constraint 5: Capacity	PCS (i=4)																						
PCS can have 2 pods	PCS (i=5)																						
per presentation	PCS (i=6)																						
(due to Chumash capacity)	PCS (i=10)																						
	PCS (i=11)																						
	PCS (i=12)																						
Constraint 6: Capacity	DSDR(i=7)																						
DSDR can have 4 pods	DSDR(i=8)																						
per presentation	DSDR(i=13)																						
(due to Rec Ctr capacity)	DSDR(i=14)																						
Constraint 7: Conflict	Group 1																						
If any Group is scheduled for	Group 2																						
Pres. i=4, they can't be in	Group 3																						
Pres. i=7, and vice versa	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 8: Conflict	Group 1																						
If any Group is scheduled for	Group 2																						
Pres. i=5, they can't be in	Group 3																						
Pres. i=8, and vice versa	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 9: Conflict	Group 1																						
If any Group is scheduled for	Group 2																						
Pres. i=10, they can't be in	Group 3																						
Pres. i=14, and vice versa	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 10: Gone because no pres i=9, so no conflict with pres i=13																							

X3,1	X3,2	X3,3	X3,4	X3,5	X3,6	X3,7	X3,8	X3,9	X3,10	X3,11	X4,1	X4,2	X4,3	X4,4	X4,5	X4,6	X4,7	X4,8	X4,9	X4,10	X4,11	X5,1	X5,2	X5,3	X5,4	X5,5	X5,6	X5,7	X5,8	X5,9	X5,10	X5,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1											1										
												1											1									
													1											1								
														1											1							
															1											1						
																1											1					
																	1											1				
																		1											1			
																			1											1		
																				1											1	
																					1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
											1																					
												1																				
													1																			
														1																		
															1																	
																1																
																	1															
																		1														
																			1													
																				1												
																					1											
																						1										
																							1									
																								1								
																									1							
																										1						
																											1					
																												1				
																													1			
																														1		
																															1	
																																1

X6,1	X6,2	X6,3	X6,4	X6,5	X6,6	X6,7	X6,8	X6,9	X6,10	X6,11	X7,1	X7,2	X7,3	X7,4	X7,5	X7,6	X7,7	X7,8	X7,9	X7,10	X7,11	X8,1	X8,2	X8,3	X8,4	X8,5	X8,6	X8,7	X8,8	X8,9	X8,10	X8,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1											1										
												1											1									
													1											1								
														1											1							
															1											1						
																1											1					
																	1											1				
																		1											1			
																			1											1		
																				1											1	
																					1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
											1																					
												1																				
													1																			
														1																		
															1																	
																1																
																	1															
																		1														
																			1													
																				1												
																					1											
																						1										
																							1									
																								1								
																									1							
																										1						
																											1					
																												1				
																													1			
																														1		
																															1	
																																1

X10,1	X10,2	X10,3	X10,4	X10,5	X10,6	X10,7	X10,8	X10,9	X10,10	X10,11	X11,1	X11,2	X11,3	X11,4	X11,5	X11,6	X11,7	X11,8	X11,9	X11,10	X11,11	X12,1	X12,2	X12,3	X12,4	X12,5	X12,6	X12,7	X12,8	X12,9	X12,10	X12,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						

X13,1	X13,2	X13,3	X13,4	X13,5	X13,6	X13,7	X13,8	X13,9	X13,10	X13,11	X14,1	X14,2	X14,3	X14,4	X14,5	X14,6	X14,7	X14,8	X14,9	X14,10	X14,11			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	1	1	0	1	0	0	33		
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
																						1	=	1
1											1											1	=	1
	1											1										1	=	1
		1											1									1	=	1
			1											1								1	=	1
				1											1							1	=	1
					1											1						1	=	1
						1											1					1	=	1
							1											1				1	=	1
								1											1			1	=	1
									1											1		1	=	1
										1											1	1	=	1
																						4	<=	4
																						4	<=	4
																						3	<=	4
																						2	<=	2
																						2	<=	2
																						2	<=	2
																						2	<=	2
																						1	<=	2
																						2	<=	2
																						3	<=	4
																						0	<=	4
1	1	1	1	1	1	1	1	1	1	1												4	<=	4
											1	1	1	1	1	1	1	1	1	1	1	4	<=	4
																						1	<=	1
																						1	<=	1
																						1	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						1	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
																						0	<=	1
											1											0	<=	1
												1										0	<=	1
													1									1	<=	1
														1								0	<=	1
															1							1	<=	1
																1						1	<=	1
																	1					1	<=	1
																		1				0	<=	1
																			1			1	<=	1
																				1		0	<=	1
																					1	1	<=	1

Appendix E: Excel Spreadsheet for Solution 3

Variables		X1,1	X1,2	X1,3	X1,4	X1,5	X1,6	X1,7	X1,8	X1,9	X1,10	X1,11	X2,1	X2,2	X2,3	X2,4	X2,5	X2,6	X2,7	X2,8	X2,9	X2,10	X2,11
Objective Function Coefficients		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Variable Place Holders		0	0	0	0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	0	0	0	0
Constraints																							
Constraint 1:	Group 1	1											1										
Each pod will see R&D	Group 2		1											1									
(i=1,2,3) once	Group 3			1											1								
	Group 4				1											1							
	Group 5					1											1						
	Group 6						1											1					
	Group 7							1											1				
	Group 8								1											1			
	Group 9									1											1		
	Group 10										1											1	
	Group 11											1											1
Constraint 2:	Group 1																						
Each pod will see PCS	Group 2																						
(i=4,5,6,9,10,11,12) once	Group 3																						
	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 3:	Group 1																						
Each pod will see DSDR	Group 2																						
(i=7,8,13,14) once	Group 3																						
	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 4: Capacity	R&D (i=1)	1	1	1	1	1	1	1	1	1	1	1											
R&D can have 4 pods	R&D(i=2)												1	1	1	1	1	1	1	1	1	1	1
per presentation	R&D(i=3)																						
Constraint 5: Capacity	PCS (i=4)																						
PCS can have 2 pods	PCS (i=5)																						
per presentation	PCS (i=6)																						
	PCS (i=9)																						
	PCS (i=10)																						
	PCS (i=11)																						
	PCS (i=12)																						
Constraint 6: Capacity	DSDR(i=7)																						
DSDR can have 4 pods	DSDR(i=8)																						
per presentation	DSDR(i=13)																						
(due to Rec Ctr capacity)	DSDR(i=14)																						
Constraint 7: One Per Day	Group 1																						
On Saturday, can only see	Group 2																						
one presentation per day	Group 3																						
(no need for Friday, b/c only	Group 4																						
R&D on Friday)	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 8: One Per Day	Group 1																						
Can only see one	Group 2																						
presentation on Sunday	Group 3																						
	Group 4																						
	Group 5																						
	Group 6																						
	Group 7																						
	Group 8																						
	Group 9																						
	Group 10																						
	Group 11																						
Constraint 9: Baseline Min.	PCS (i=4)																						
PCS presentations need	PCS (i=5)																						
at least 1 pod per presentation	PCS (i=6)																						
	PCS (i=9)																						
	PCS (i=10)																						
	PCS (i=11)																						
	PCS (i=12)																						
Constraint 10: Baseline Min.	DSDR(i=7)																						
DSDR presentations need	DSDR(i=8)																						
at least 2 pods per pres	DSDR(i=13)																						
	DSDR(i=14)																						

X3,1	X3,2	X3,3	X3,4	X3,5	X3,6	X3,7	X3,8	X3,9	X3,10	X3,11	X4,1	X4,2	X4,3	X4,4	X4,5	X4,6	X4,7	X4,8	X4,9	X4,10	X4,11	X5,1	X5,2	X5,3	X5,4	X5,5	X5,6	X5,7	X5,8	X5,9	X5,10	X5,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1											1										
												1											1									
													1											1								
														1											1							
															1											1						
																1											1					
																	1											1				
																		1											1			
																			1											1		
																				1											1	
																					1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
											1											1										
												1											1									
													1											1								
														1											1							
															1											1						
																1											1					
																	1											1				
																		1											1			
																			1											1		
																				1											1	
																					1											1
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1

X6,1	X6,2	X6,3	X6,4	X6,5	X6,6	X6,7	X6,8	X6,9	X6,10	X6,11	X7,1	X7,2	X7,3	X7,4	X7,5	X7,6	X7,7	X7,8	X7,9	X7,10	X7,11	X8,1	X8,2	X8,3	X8,4	X8,5	X8,6	X8,7	X8,8	X8,9	X8,10	X8,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	1	0
1																																
	1																															
		1																														
			1																													
				1																												
					1																											
						1																										
							1																									
								1																								
									1																							
										1																						
											1											1										
												1											1									
													1											1								
														1											1							
															1											1						
																1											1					
																	1											1				
																		1											1			
																			1											1		
																				1											1	
																					1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1

X9,1	X9,2	X9,3	X9,4	X9,5	X9,6	X9,7	X9,8	X9,9	X9,10	X9,11	X10,1	X10,2	X10,3	X10,4	X10,5	X10,6	X10,7	X10,8	X10,9	X10,10	X10,11	X11,1	X11,2	X11,3	X11,4	X11,5	X11,6	X11,7	X11,8	X11,9	X11,10	X11,11
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1
1											1											1										
	1											1											1									
		1											1											1								
			1											1											1							
				1											1											1						
					1											1											1					
						1											1											1				
							1											1											1			
								1											1											1		
									1											1											1	
										1											1											1
1	1	1	1	1	1	1	1	1	1	1																						
											1	1	1	1	1	1	1	1	1	1	1											
																						1	1	1	1	1	1	1	1	1	1	1

X12,1	X12,2	X12,3	X12,4	X12,5	X12,6	X12,7	X12,8	X12,9	X12,10	X12,11	X13,1	X13,2	X13,3	X13,4	X13,5	X13,6	X13,7	X13,8	X13,9	X13,10	X13,11	X14,1	X14,2	X14,3	X14,4	X14,5	X14,6	X14,7	X14,8	X14,9	X14,10	X14,11			
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			
0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	33		
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
1																																	1	=	1
	1																																1	=	1
		1																															1	=	1
			1																														1	=	1
				1																													1	=	1
					1																												1	=	1
						1																											1	=	1
							1																										1	=	1
								1																									1	=	1
									1																								1	=	1
										1																							1	=	1
											1											1											1	=	1
												1											1										1	=	1
													1											1									1	=	1
														1											1								1	=	1
															1											1							1	=	1
																1											1						1	=	1
																	1											1					1	=	1
																		1											1				1	=	1
																			1											1			1	=	1
																				1											1		1	=	1
																					1											1	1	=	1
																																	4	<=	4
																																	4	<=	4
																																	3	<=	4
																																	1	<=	2
																																	2	<=	2
																																	2	<=	2
																																	2	<=	2
																																	1	<=	2
																																	2	<=	2
1	1	1	1	1	1	1	1	1	1	1																							1	<=	2
																																	4	<=	4
																																	2	<=	4
											1	1	1	1	1	1	1	1	1	1	1												3	<=	4
																						1	1	1	1	1	1	1	1	1	1	1	2	<=	4
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
																																	1	=	1
1											1											1											1	=	1
	1											1											1										1	=	1
		1											1											1									1	=	1
			1											1											1								1	=	1
				1											1											1							1	=	1
					1											1											1						1	=	1
						1											1											1					1	=	1
							1											1											1				1	=	1
								1											1											1			1	=	1
									1											1											1		1	=	1
										1											1											1	1	=	1
																																	1	>=	1
																																	2	>=	1
																																	2	>=	1
																																	2	>=	1
																																	1	>=	1
																																	2	>=	1
1	1	1	1	1	1	1	1	1	1	1																							1	>=	1
																																	4	>=	2
																																	2	>=	2
											1	1	1	1	1	1	1	1	1	1	1												3	>=	2
																						1	1	1	1	1	1	1	1	1	1	1	2	>=	2

Appendix F: Alternative Solution Summary and Comparison

	Number of Presentations (i)	Number of Pods (j)	Number of Constraints	Number of Pods w/Same Day Presentations	Details
Solution 1	14	11	91	3	Basic solution, no extra constraints
Solution 2a	13	11	90	4	Removed Presentation i=12, PCS Sun 3-4PM i=8 not utilized, DSDR Sat 12-1PM
Solution 2b	13	11	90	4	Removed Presentation i=11, PCS Sun 1:30-2:30PM i=8 not utilized, DSDR Sat 12-1PM
Solution 2c	13	11	79	4	Removed Presentation i=9, PCS Sun 10-11AM i=8 not utilized, DSDR Sat 12-1PM
Solution 3	14	11	80	0	Removed Time Conflict Constraints All presentations utilized Demonstrates "See Only One Pres A Day" Constraints Introduces baseline minimum constraints (in addition to a maximum constraint i.e. each DSDR has a max of 4 and a min of 2) R&D doesn't need such a constraint, already at max.

Appendix G: Pugh Chart Weighing Each of the Potential Solutions

System Name		Previous System	Basic	Removed a presentation offering			Different Constraints
Additional Details		Done by hand, no particular system to assigning groups to presentations	Original solution with all constraints, i=8 (DSDR Sat 12-1PM) and i=11 (PCS Sun 1:30-2:30PM not utilized	Removed presentation i=12 (PCS Sun 3-4PM), i=8 (DSDR Sat 12-1PM)not utilized	Removed presentation i=11 (PCS Sun 1:30-2:30PM), i=8 (DSDR Sat 12-1PM) not utilized	Removed presentation i=9 (PCS Sun 10-11AM), i=8 (DSDR Sat 12-1PM) not utilized	Removed time conflict constraints, and utilized "See One Presentation a Day" constraints, all presentaitons utilized
Criteria	Weight	DATUM	Solution 1	Solution 2a	Solution 2b	Solution 2c	Solution 3
Variable Flexibility	3	0	0	+	+	+	0
Constraint Flexibility	3	0	0	0	0	0	+
Presentation Schedule Organization	2	0	0	-	-	-	+
Likelihood of committing errors	2	0	+	+	+	+	+
Presentations Sensitive to Overcapacity	1	0	+	-	-	-	++
Ease of Understanding	3	0	-	-	-	-	-
Use of Time/Energy/Resources	1	0	0	+	+	+	0
+		0	3	6	6	6	9
0		11	9	3	3	3	4
-		0	3	6	6	6	3
Net Score		0	0	0	0	0	6