# Optimizing Academic Timetables Using Integer Linear Programming: A Case Study

Irit Talmor[*]

Sir Harry Solomon School of Management, Western Galilee College, POB 2125 Akko, Israel

[*] Correspondence: Irit Talmor (iritt@wgalil.ac.il)

**Abstract:** An Integer Linear Programming (ILP) model was proposed to optimize academic timetables, with a focus on the School of Management at Western Galilee Academic College. The model was designed to address the dual challenges of course scheduling and faculty availability while incorporating a minor structural adjustment to enhance computational efficiency and accelerate convergence, particularly for large-scale problems. By employing this model, optimal scheduling solutions were generated within minutes, even in scenarios involving over 200 classes and 100 lecturers. The approach effectively minimizes planning time, identifies unavoidable scheduling conflicts, and highlights unschedulable classes due to constraint violations. Furthermore, the model provides actionable insights into staffing requirements, ensuring a comprehensive resource allocation strategy. Results from the application of the model during the 2023 winter semester demonstrated its capability to efficiently schedule 236 classes across multiple programs and instructional modalities. The method achieved adherence to predefined constraints, optimized the utilization of institutional resources, and enhanced overall scheduling efficiency. This case study underscores the potential of the proposed ILP framework to streamline academic timetabling processes, particularly in environments with diverse programmatic needs and complex resource interdependencies. The findings indicate that the model can be readily adapted to other academic institutions seeking to improve the effectiveness and precision of their scheduling systems.

**Keywords:** University timetable; Lecturers' availability; Assignment problems; Course scheduling; Optimality

## 1. Introduction

The concept of assignment has broadened significantly from its initial definition of distributing tasks among workers or machines in various organizations. It now encompasses a wide range of strategic decisions, from determining the locations of branches and service centers to allocating specialized equipment across different sites and addressing recruitment needs. Making optimal assignment decisions involves maximizing the benefits and/or minimizing the costs associated with these activities.

One common application of the assignment problem is the timetabling challenge, which is vital in guaranteeing the regular functioning of sectors such as education (Lee & Schniederjans, 1983; Tan et al., 2021), healthcare (Bilgin et al., 2012; Nossack, 2022; Yinusa & Faezipour, 2023), transportation (Daş et al., 2020; Valouxis & Housos, 2002), and many other fields. In this context, the problem of setting academic timetables received particular attention over the years. The three main factors applicable to this sub-group of problems are course timetabling, classroom usage and faculty.

Course timetabling is a classic problem and one of the most popular and closely studied in the research literature. It refers to setting the weekly timetable of courses, taking into account complex considerations such as avoiding overlap and fluctuations in student enrolment (Abuhamdah et al., 2014; Aladağ & Hocaoglu, 2007; Al-Betar & Khader, 2012; Burke & Petrovic, 2002; Chen & Shih, 2013; Di Gaspero & Schaerf, 2006; Faudzi et al., 2018; Lü & Hao, 2010; Song et al., 2018; Soria-Alcaraz et al., 2016). Some studies also combined soft constraints in their methodology, such as minimizing transition time between classes (Abuhamdah & Ayob, 2009) or maximizing student preferences (Abuhamdah & Ayob, 2009; Colajanni & Daniele, 2021).

Classroom assignment refers to the problem of allocating classrooms to courses that have already been scheduled, while considering classroom capacity, the movement of students between classes, and so on (Burke & Petrovic, 2002; Dimopoulou & Miliotis, 2001; Faudzi et al., 2018; Lemos et al., 2019; Oladejo et al., 2019; Phillips

et al., 2017). A branch of this sub-group is the challenge of assigning classrooms for final exams, which should fulfill unique requirements, such as intervals between exams and strict limitations on the number of students per class (Dimopoulou & Miliotis, 2001).

The third component of the academic timetabling sub-group is faculty. That is, assigning lecturers and teaching assistants to courses based on their qualifications, preferences and availability, while taking into account the employment status of different faculty members (Domenech & Lusa, 2016).

Some previous studies integrated scheduling both course timetables and classroom usage (Daskalaki et al., 2004; Murray et al., 2007) or both course timetables and faculty assignments (Arratia-Martinez et al., 2021; MirHassani, 2006). However, these models are less common.

There are various approaches for solving university course timetabling problems (UCTP). The most common is the accurate solution approach, which involves mixed-integer and binary-linear programming (Arratia-Martinez et al., 2021; Colajanni & Daniele, 2021; Daskalaki et al., 2004; Dimopoulou & Miliotis, 2001; Domenech & Lusa, 2016; MirHassani, 2006; Mokhtari et al., 2021; Oladejo et al., 2019; Phillips et al., 2015). For problems that are too large to converge to an optimal solution within a reasonable time, approximate methods and efficient heuristics are suggested, such as local search (Abuhamdah et al., 2014; Ceschia et al., 2014; Chen & Shih, 2013; Song et al., 2018; Soria-Alcaraz et al., 2016), tabu search (Aladağ & Hocaoğlu, 2007; Lü & Hao, 2010), practical collision algorithm (Abuhamdah & Ayob, 2009), harmony search algorithm (Al-Betar & Khader, 2012), and greedy algorithm (Lemos et al., 2019). Comprehensive surveys of UCTP and their solution methods can be found in the studies by Burke & Petrovic (2002), Faudzi et al. (2018), and Herres & Schmitz (2021) and in recent studies by Abdipoor et al. (2023), Bashab et al. (2023), Ceschia et al. (2023), Mallari et al. (2023), and Müller et al. (2024).

Most studies addressing the course timetabling problem, regardless of employing precise or approximate methods, typically do not aim to develop a universal solution. Instead, each study presents a customized model crafted to address the specific needs of a particular department at one institution. Examples of such studies include the following: Murray et al. (2007) developed a timetabling solution for Purdue University; Lemos et al. (2019) focused on the Instituto Superior Técnico (IST) at the University of Lisbon; Mokhtari et al. (2021) created a schedule for postgraduate courses at the Industrial Engineering Department of Islamic Azad University, Najafabad Branch; and Colajanni & Daniele (2021) custom-tailored a timetable for first-year mathematics students at the University of Catania, Italy.

A highly relevant study by Arratia-Martinez et al. (2021) integrates several features discussed above. It utilizes a binary linear programming approach to construct a lecturer-course-time slot timetable for a specific department at a Mexican academic institution. Beyond the standard constraints, the model incorporates various characteristics such as the lecturers' positions, the number of courses assigned to each one based on their academic profile, and the list of courses each lecturer is qualified to teach. The researchers applied their model to a case study involving 29 lecturers, 24 courses and 69 time slots.

Building on this approach, the current study introduces an enhanced ILP model for developing an academic timetable for the School of Management at Western Galilee Academic College. While addressing a similar problem of course scheduling and staff availability, the proposed model integrates a distinctive feature that significantly accelerates the convergence to a solution. As a result, even complex scenarios involving more than 200 classes and over 100 lecturers were resolved in less than two minutes, offering a substantial improvement in efficiency.

The next sections of this study are structured as follows: the subsequent two sections outline the specific characteristics of the problem. This is followed by a detailed mathematical formulation. The final sections are devoted to the implementation of the model and discussion of its results and directions for future research.

## 2. Problem Description

The School of Management at the Western Galilee Academic College offers four academic programs: Economics and Accounting, Economics and Management, Management, and Management and Logistics. Besides Economics and Accounting, which is designed for completion within four years, all other programs are designed for completion within three years. The programs are available in various modalities, including morning, evening, or both. In total, the school manages 25 distinct combinations of program, year of study and modality (hereafter – curriculum).

In order to accommodate working individuals, the school has a policy of dedicating two to three study days per week to each program combination. Although these designated days may vary annually, once established, they apply to all courses included in any specific combination. This means that every session of each course within the combination is scheduled exclusively during a time slot (specific weekday and timeframe) that aligns with the combination's study days and modality. Table 1 presents the programs, their modalities, and their study days. Note that there are no classes on Thursday.

Courses at the school are offered per semester. Each course is typically assigned one weekly time slot. However, courses comprising both a lecture and a tutorial receive two time slots per week, one for each component. For

high-enrolment courses, the school sets up multiple tutorials. The total number of courses offered each semester varies between 120 and 160. Some of them include tutorials, resulting in approximately 200 to 250 classes that need to be scheduled.

**Table 1.** Modality and study days of each program

| Program | Modality | Year of Study | Studying Days | Symbol |
|---|---|---|---|---|
| Economics and Accounting | Morning | First | Sun, Tue, Fri | EA_M_1 |
| | | Second | Sun, Tue, Fri | EA_M_2 |
| | | Third | Sun, Mon, Tue | EA_M_3 |
| | Evening | First | Sun, Tue, Fri, | EA_E_1 |
| | | Second | Mon, Wed, Fri | EA_E_2 |
| | | Third | Sun, Tue, Fri | EA_E_3 |
| | | Third | Mon, Wed, Fri | EA_E_4 |
| Economics and Management | Morning | First | Sun, Mon, Tue | EM_M_1 |
| | | Second | Sun, Mon, Tue | EM_M_2 |
| | | Third | Sun, Mon, Wed | EM_M_3 |
| | Evening | First | Sun, Tue, Fri | EM_E_1 |
| | | Second | Mon, Wed, Fri | EM_E_2 |
| | | Third | Sun, Tue, Fri | EM_E_3 |
| Management | Morning | First | Sun, Mon, Tue | M_M_1 |
| | | Second | Sun, Mon, Wed | M_M_2 |
| | | Third | Sun, Mon, Wed | M_M_3 |
| | Evening | First | Sun, Tue, Fri | M_E_1 |
| | | Second | Mon, Wed, Fri | M_E_2 |
| | | Third | Sun, Tue, Fri | M_E_3 |
| | Both morning &evening | First | Sun, Fri | M_1 |
| | | Second | Sun, Fri | M_2 |
| | | Third | Mon, Fri | M_3 |
| Management and Logistics | Evening | First | Sun, Tue, Fri | ML_E_1 |
| | | Second | Mon, Wed, Fri | ML_E_2 |
| | | Third | Sun, Tue, Fri | ML_E_3 |

The teaching staff at the school comprises two groups: faculty and non-faculty. The latter group includes teaching assistants and non-faculty lecturers. Generally, members of both groups are eligible to teach courses based on their expertise, and the specific assignments they get are based on their availability. The key difference between them is that faculty members must be assigned to teach courses, whereas non-faculty members are only assigned teaching duties if needed. The total number of lecturers typically ranges from 70 to 100 (comprising members of both groups).

Several months before the start of the academic year, administrators at the School of Management undertake the complex task of constructing timetables for each curriculum. Their primary objective is to schedule all required courses while adhering to predefined conditions. Currently, this process is done manually using paper and pencil, which is time-consuming, often taking weeks to complete. Moreover, the predefined constraints are often inadvertently overlooked or disregarded. A significant challenge for administrators is distinguishing between unavoidable lapses and those caused by poor planning, as they lack the means to evaluate their solutions.

The model and its application, as presented in this study, assist administrators in three critical ways: first, it significantly reduces planning time to just a few minutes; second, it easily identifies inevitable scheduling conflicts and highlights any classes that cannot be scheduled due to predefined constraints; third, it provides clear insights into staffing requirements for lecturers. The detailed specified list of constraints is presented in Section 3 below.

## 3. Problem Specifications

The problem includes four main specifications, as follows:

a) Set of curriculums. Each curriculum represents a unique instance of "program + year of study + modality".

b) Set of academic courses. Each course is exclusively associated with one curriculum, and the number of classes for each course is known. The following points should be emphasized:

• Courses with lectures only: Courses consisting solely of lectures are conducted according to the number of available classes, which is determined by the number of enrolled students.

• Courses with lectures and tutorials: Some courses include both lectures and tutorials. In these cases, multiple tutorial sessions may be offered, contingent upon the number of enrolled students.

• Parallel scheduling of courses: Courses within the same curriculum can be scheduled in parallel only if they are offered in multiple classes.

c) Time slots: This is a collection of pairs (day of the week, timeframe) in which courses can be scheduled.

Classes take place from Sunday to Friday (inclusive, with Thursdays off). To simplify the problem, the timeframes are uniformly defined as follows:
  • Sunday-Wednesday:
Morning modality sessions: 8:00-10:30, 10:30-13:00, 13:00-15:30
Evening modality sessions: 18:00-20:30, 20:30-23:00
Timeframe available for both morning and evening modality sessions: 15:30-18:00
  • Friday sessions: 08:00-10:30, 10:30-13:00, 13:00-15:30

For example, consider a curriculum with a morning modality on Sundays and Fridays. This curriculum would include seven time slots: 8:00-10:30 on Sunday, 10:30-13:00 on Sunday, 13:00-15:30 on Sunday, 15:30-18:00 on Sunday, 08:00-10:30 on Friday, 10:30-13:00 on Friday, and 13:00-15:30 on Friday. All classes of courses included in this curriculum must be scheduled within these time slots, with each class assigned to a single time slot. It's important to note that while a course is associated with only one curriculum, a time slot may be shared by multiple curriculums.

d) Set of lecturers: This set comprises each lecturer and a list of courses that they are eligible to teach. It also details their availability on specific weekdays and their employment status expressed as a percentage, indicating whether they hold a part-time or full-time position.

## 4. Mathematical Formulation

### 4.1 Decision Variables

In the context of this study, decision variables were categorized into two distinct types:

a) Binary variables $x_{ijt}$: Each binary variable represents the allocation of a specific class in a course to a designated time slot, taught by a particular lecturer.

$$x_{ijt} = \begin{cases} 1 & \begin{array}{c} if\ lecturer\ i\ teaches\ a\ class\ of\ course\ j \\ at\ time_{slot}t \end{array} \\ 0 & o.w. \end{cases}$$

b) Continuous assisting variables $y_{jt}$: Each variable represents the allocation of a specific course to a designated time slot. That is, the value of $y_{jt}$ is greater than 0 only if at least one class of course $j$ is scheduled at time slot $t$. The connection between $y_{jt}$ and $x_{ijt}$ is formulated in Eqs. (4) and (5), linking scheduling of course-lecturer-time slots with overall course scheduling.

### 4.2 Objective Function

The primary objective of schedule planners is to maximize the total number of allocated classes. This objective is formulated in Eq. (1), as follows:

$$Max \sum_{i \in All\_teachers} \sum_{j \in All\_courses} \sum_{t \in All\_t\_slots} x_{ijt} \tag{1}$$

### 4.3 Constraints

The constraints within the model are divided into two primary categories: course constraints and lecturer constraints.

4.3.1 Course constraints
a) A course included in curriculum $\Psi$ can only be assigned to a time slot allocated to its curriculum.

$$y_{jt} = 0 \quad \forall j \in courses(\Psi), t \notin T(\Psi), \qquad \forall \Psi \tag{2}$$

b) The total number of classes for course $j$ must not surpass the required number of classes for that course.

$$\sum_{t \in T(\Psi)} y_{jt} \leq N_j \quad \forall j \in courses(\Psi), \forall \Psi \tag{3}$$

c) Two interconnected constraints regulate the assignment of course classes within specific time slots and the course's overall scheduling: Eq. (4) ensures that $y_{jt} > 0$ if at least one class of course $j$ is scheduled in time slot

$t$, indicating that the course has been scheduled. Eq. (5) asserts that $y_{jt} = 0$ if no classes of course $j$ are scheduled in time slot $t$.

$$y_{jt} \geq \sum_{\forall i \in All\_teachers} x_{ijt}/N_j \quad \forall j \in courses(\Psi), t \in T(\Psi), \quad \forall \Psi \tag{4}$$

$$y_{jt} \leq \sum_{\forall i \in All\_teachers} x_{ijt} \quad \forall j \in courses(\Psi), t \in T(\Psi), \quad \forall \Psi \tag{5}$$

d) The subsequent constraints, together with Eqs. (4) and (5), were designed to prevent scheduling conflicts for courses within the same curriculum, as follows: If all classes of a course are allocated to a single time slot, no other course within the same curriculum may be assigned to this time slot; on the other hand, for courses comprising multiple classes that are offered at several time slots, overlapping assignments are permitted. Those three sets of constraints guarantee that students enrolled in this curriculum have the ability to register for all their required courses without any conflict.

$$\sum_{j \in courses(\Psi)} y_{jt} \leq 1 \quad \forall t \in T(\Psi), \quad \forall \Psi \tag{6}$$

### 4.4.2 Lecturer constraints
a) Lecturers can only be assigned to courses that they are qualified to teach:

$$x_{ijt} = 0 \quad \forall i \in All_{teachers}, j \notin C(i), \quad \forall \Psi \tag{7}$$

b) A lecturer can teach no more than one class in each time slot:

$$\sum_{j \in All\_courses} x_{ijt} \leq 1 \quad \forall i \in All_{teachers}, \quad \forall t \in All\_t\_slots \tag{8}$$

c) A lecturer must be assigned teaching hours that align with their defined employment status:

$$\sum_{j \in All_{courses}} h_j \sum_{t \in All_{t_{slots}}} x_{ijt} \geq Min\_position_i \quad \forall i \in All_{teachers} \tag{9}$$

$$\sum_{j \in All_{courses}} h_j \sum_{t \in All_{t_{slots}}} x_{ijt} \leq Max_{position_i} \quad \forall i \in All_{teachers} \tag{10}$$

## 4.4 Model Coding

The model was coded with Python version 3.7.2 and run on a standard personal laptop (processor: Intel i7-7500U CPU @ 2.70GHz, x64, 16.0GB RAM), using the CBC engine solver. The performance described hereafter refers to this hardware.

## 5. Implementation

### 5.1 Input Data

The school of management had to assign 236 classes for 201 courses in the 2023 winter semester, as follows: 146 courses had one lecture per week; 24 courses had one lecture plus one tutorial per week; 35 courses included one lecture plus two tutorials per week (students can choose which of the two tutorials to attend). It should be noted, however, that similar courses taught in different programs are treated as separate courses. The results presented hereafter focus on this semester because scheduling demands in other semesters were simpler (Table 2).

The lecturer pool included 25 faculty members and 75 non-faculty lecturers and teaching assistants. Each faculty member is obligated to teach a certain number of hours in line with their contract. The positions of non-faculty teachers were set according to their teaching assignments, up to the maximal load permitted to them. Detailed anonymized data is presented in Table A1 and Table A2 of the appendix.

Table 2. Course map

| Semesters | Lectures | Lectures with One Tutorial | Lectures with Two Tutorials | Total Number of Courses | Total Number of Classes |
|---|---|---|---|---|---|
| Winter | 146 | 24 | 35 | 201 | 236 |
| Spring | 123 | 44 | 13 | 180 | 193 |
| Summer | 29 | 10 | 0 | 39 | 39 |

## 5.2 Results

Given the complexity of the problem, the model was applied iteratively to the input data, progressively imposing restrictive conditions. The process was conducted in the following stages:

**Initial run:** The first run aimed to verify the existence of a solution and ensure all classes could be scheduled. For this, two basic restrictions were loosened: the lower-limit positions were set to 0, and the availability of all lecturers was extended to all weekdays. Under these assumptions, the model included 306,939 variables and 407,886 constraints. A solution was obtained after a runtime of 1.84 minutes, successfully scheduling all 236 classes. This solution utilized 84 lecturers, 20 of whom were assigned to only one class.

**Second run:** Based on the initial run's output, the input data was updated: 16 non-faculty lecturers who were not assigned any classes were removed from the list; the lower-limit positions for faculty members were set to a minimal positive value – 1 point. The model converged to an optimal solution: all 236 classes were scheduled utilizing 80 lecturers, with 15 of them assigned to only one class. The run duration was 1.48 minutes.

**Third run:** Based on the second run's output, more restrictions were imposed on the input data: the total number of lecturers was reduced to 80 (25 faculty, 55 non-faculty); the availability of faculty members was updated to three days a week, and their minimal teaching commitment was updated to 3 points. This run converged to an optimal solution within 1.49 minutes. All 236 classes were scheduled, and 76 lecturers were assigned, with nine of them teaching only one class.

**Final run:** The model was run again, with the same input as the previous run, except for four unassigned lecturers who were removed from the list. Thus, 234,003 variables and 311,421 constraints were included in the problem. The run converged into a solution within 1.45 minutes with all 76 lecturers on the list assigned to classes. This outcome satisfied the goals and therefore the process was ended at this point.

Table 3 summarizes the solution process. It can be noted that the working day distribution was changed between the third and final run. This result indicates that several suitable timetables were available. In this case, the timetable of the final run was chosen to reduce the number of workdays a week for non-faculty lecturers.

Table 3. Solution convergence process

| | Run Stage | Initial | | Second | | Third | | Final | |
|---|---|---|---|---|---|---|---|---|---|
| | Available lecturers | 100 | | 84 | | 80 | | 76 | |
| Input data [1] | $F^{[2]} + NF^{[3]}$ | 25+75 | | 25+59 | | 25+55 | | 25+51 | |
| | Lower limit teaching points for faculty | 0 | | 1 | | 3 | | 3 | |
| | Number of variables | 306,939 | | 258,315 | | 246,159 | | 234,003 | |
| | Number of constraints | 407,886 | | 343,266 | | 327,513 | | 311,421 | |
| | Run duration (min.) | 1.84 | | 1.48 | | 1.49 | | 1.45 | |
| | Solution found? | Yes. All 236 classes were scheduled. | | | | | | | |
| | Lecturers assigned | 84 | | 80 | | 76 | | 76 | |
| | Teaching days p/w | F | NF | F | NF | F | NF | F | NF |
| | 1 | 19 | 5 | 2 | 4 | 4 | 10 | 6 | 14 |
| General output | 2 | 25 | 8 | 12 | 28 | 13 | 25 | 10 | 23 |
| | 3 | 10 | 8 | 9 | 15 | 8 | 11 | 9 | 9 |
| | 4 | 5 | 4 | 2 | 8 | 0 | 4 | 0 | 5 |
| | 5 | | | | | | | 0 | 1 |

(1) Basic input data is common to all runs of the course map, as shown in Table 2. The upper limit of teaching points for faculty is based on employment status (up to 12). The lower and upper limits of teaching points for non-faculty are 0 and 12, respectively.
(2) F indicates faculty.
(3) NF indicates non-faculty.

Table 4 shows the distribution of courses during the week depending on the program for the chosen timetable, which is based on the output of the final run. Notably, the iterative process resulted in quick convergence to a solution that adhered to all constraints and saved resources. In addition, because almost all programs included Sunday as a school day, the number of classes on Sunday is the highest of 81. On the other hand, Wednesday was

**Table 4.** Class distribution during the week for each program in the final run

| Program | Symbol | Sun | Mon | Tue | Wed | Fri |
|---|---|---|---|---|---|---|
| | EA_M_1 | 6 | | 6 | | 4 |
| | EA_M_2 | 4 | | 4 | | 3 |
| | EA_M_3 | 3 | 3 | 4 | | |
| Economics and Accounting | EA_M_4 | | 3 | 1 | | 1 |
| | EA_E_1 | 3 | | 3 | | 3 |
| | EA_E_2 | | 2 | | 3 | 3 |
| | EA_E_3 | 3 | | 3 | | 3 |
| | EM_M_1 | 6 | 5 | 4 | | |
| | EM_M_2 | 7 | 4 | 4 | | |
| Economics and Management | EM_M_3 | | 2 | | 1 | 3 |
| | EM_E_1 | 5 | | 4 | | 4 |
| | EM_E_2 | | 5 | | 2 | 3 |
| | EM_E_3 | 1 | | 2 | | 3 |
| | M_M_1 | 5 | 4 | 5 | | |
| | M_M_2 | 3 | 4 | | 2 | |
| | M_M_3 | 3 | 2 | | 2 | |
| | M_E_1 | 6 | | 3 | | 3 |
| Management | M_E_2 | | 3 | | 3 | 2 |
| | M_E_3 | 3 | | 2 | | 2 |
| | M_1 | 9 | | | | 4 |
| | M_2 | 7 | | | | 3 |
| | M_3 | | 5 | | | 3 |
| | ML_E_1 | 2 | | 3 | | 1 |
| Management and Logistics | ML_E_2 | | 2 | | 3 | 3 |
| | ML_E_3 | 3 | | 3 | | 1 |
| Total | | 81 | 43 | 48 | 16 | 48 |

When examining the class distribution across time slots, Figure 1 shows that the load is almost evenly spread throughout the day. The exception is the 15:30-18:00 slot, where the number of scheduled courses is roughly double compared to other timeframes. This is the most popular time slot as it was offered in both morning and evening programs.
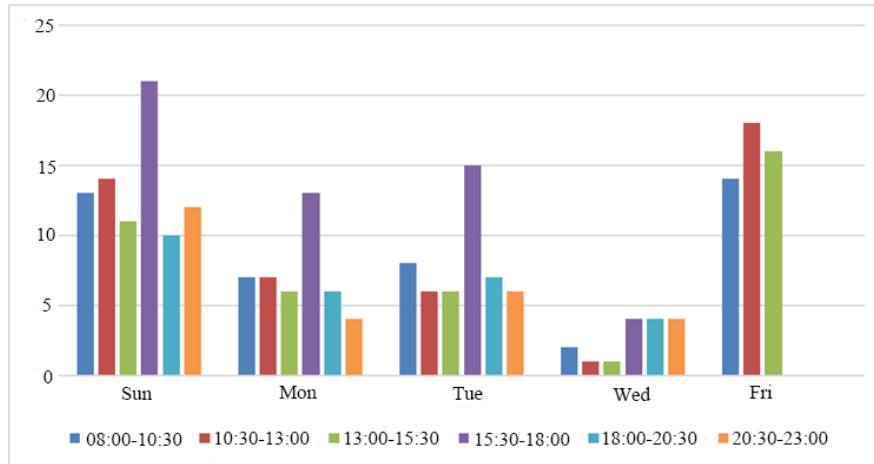


**Figure 1.** Class distribution along timeslots

An alternative approach to applying the model involves defining the availability of lecturers in advance and identifying gaps in course assignments. To illustrate this option, the input data from the previous (final) run was updated by specifying that each faculty member is available only on Sundays and Wednesdays (hereafter referred to as "limited availability"). Table 5 shows the comparison of the results. The number of courses assigned to faculty members decreased by 16, while the number of courses assigned to non-faculty lecturers decreased by only 9, leaving 7 unscheduled courses. The pattern of course distribution on weekdays and teaching load was unchanged. These results suggest that, given the basic input provided, it is feasible to create a complete timetable that accommodates both program requirements and lecturer preferences.

**Table 5.** A comparison of the number of classes assigned to each group of lecturers

| Lecturer Group | Final | Limited Availability |
|---|---|---|
| Faculty | 84 | 68 |
| Non-faculty | 152 | 161 |
| Not assigned | 0 | 7 |

## 6. Conclusions and Future Work

The ILP model developed for optimizing academic timetables at the School of Management, Western Galilee Academic College, demonstrates significant improvements in planning efficiency, constraint adherence, and resource allocation compared to the manual planning approach. By incorporating a minor adjustment, the model dramatically accelerated the convergence process, enabling the scheduling of over 200 classes and more than 100 lecturers within minutes. The model reduced planning time, identified inevitable violations of predefined constraints, and provided valuable insights into staffing requirements.

The iterative process used to apply the model ensured that all constraints were met and resources were optimally utilized. The results from the 2023 winter semester show that the model successfully scheduled 236 classes across various programs and modalities, significantly improving the timetable creation process and overall efficiency. In contrast, the proposed model streamlined this process, requiring around ten hours for input preparation, approximately eight minutes for algorithm execution, and a total of two hours for processing results between consecutive runs—an initial improvement of over 85% in planning duration. Future applications of the model are expected to be even more efficient, as the input file can be reused with only minor adjustments, requiring no more than one additional hour. Over time, the entire end-to-end process is projected to take no more than five hours—a dramatic reduction from the original 100 hours, representing a 95% decrease in time investment.

Furthermore, the model's outcome enables a reduction in the number of non-faculty staff required for scheduling, promoting more efficient workforce utilization and substantial resource savings. These improvements underscore the model's potential for transforming the scheduling process into a highly efficient and sustainable practice.

Future work should focus on enhancing the model's flexibility and scalability. This includes handling larger datasets and, most importantly, applying it to all three semesters in parallel (winter, spring, summer) to facilitate the creation of a yearly timetable and teaching assignments in a single run. Additionally, limiting the number of hours a lecturer teaches per day is an important consideration for ensuring workload balance and maintaining teaching quality. Although this aspect falls outside the scope of the current problem formulation, it represents a valuable direction for extending the model to incorporate faculty scheduling constraints in future research.

By addressing these areas, the ILP model can evolve into a more robust and versatile tool, capable of meeting the dynamic needs of academic institutions and further enhancing the efficiency of the timetabling process.

## Data Availability

The data used to support the research findings are available from the corresponding author upon request.

## Conflicts of Interest

The author declares no conflict of interest.

## References

Abdipoor, S., Yaakob, R., Goh, S. L., & Abdullah, S. (2023). Meta-heuristic approaches for the university course timetabling problem. *Intell. Syst. Appl.*, *19*, 200253. https://doi.org/10.1016/j.iswa.2023.200253.

Abuhamdah, A. & Ayob, M. (2009). Experimental result of particle collision algorithm for solving course timetabling problems. *Int. J. Comput. Sci. Network Secur.*, *9*(9), 134-142. http://paper.ijcsns.org/07_book/200909/20090917.pdf

Abuhamdah, A., Ayob, M., Kendall, G., & Sabar, N. R. (2014). Population based Local Search for university course timetabling problems. *Appl. Intell.*, *40*, 44-53. https://doi.org/10.1007/s10489-013-0444-6.

Aladağ, Ç. H. & Hocaoğlu, G. (2007). A tabu search algorithm to solve a course timetabling problem. *Hacettepe J. Math. Stat.*, *36*(1), 53-64.

Al-Betar, M. A. & Khader, A. T. (2012). A harmony search algorithm for university course timetabling. *Ann. Oper. Res.*, *194*, 3-31. https://doi.org/10.1007/s10479-010-0769-z.

Arratia-Martinez, N. M., Maya-Padron, C., & Avila-Torres, P. A. (2021). University course timetabling problem with professor assignment. *Math. Prob. Eng.*, *2021*(1), 6617177. https://doi.org/10.1155/2021/6617177.

Bashab, A., Ibrahim, A. O., Tarigo Hashem, I. A., Aggarwal, K., Mukhlif, F., Ghaleb, F. A., & Abdelmaboud, A. (2023). Optimization techniques in university timetabling problem: Constraints, methodologies, benchmarks,

and open issues. *Comput. Mater. Continua*, *74*(3), 6461-6484. https://doi.org/10.32604/cmc.2023.034051.

Bilgin, B., Demeester, P., Misir, M., Vancroonenburg, W., & Vanden Berghe, G. (2012). One hyper-heuristic approach to two timetabling problems in health care. *J. Heuristics*, *18*, 401-434. https://doi.org/10.1007/s10732-011-9192-0.

Burke, E. K. & Petrovic, S. (2002). Recent research directions in automated timetabling. *Eur. J. Operational Res.*, *140*(2), 266-280. https://doi.org/10.1016/S0377-2217(02)00069-3.

Ceschia, S., Di Gaspero, L., & Schaerf, A. (2023). Educational timetabling: Problems, benchmarks, and state-of-the-art results. *Eur. J. Operational Res.*, *308*(1), 1-18. https://doi.org/10.1016/j.ejor.2022.07.011.

Ceschia, S., Di Gaspero, L., & Schaerf, A. (2014). The generalized balanced academic curriculum problem with heterogeneous classes. *Ann. Oper. Res.*, *218,* 147-163. https://doi.org/10.1007/s10479-013-1358-8.

Chen, R. M. & Shih, H. F. (2013). Solving university course timetabling problems using constriction particle swarm optimization with local search. *Algorithms*, *6*(2), 227-244. https://doi.org/10.3390/a6020227.

Colajanni, G. & Daniele, P. (2021). A new model for curriculum-based university course timetabling. *Optim. Lett.*, *15*(5), 1601-1616. https://doi.org/10.1007/s11590-020-01588-x.

Daş, G. S., Gzara, F., & Stützle, T. (2020). A review on airport gate assignment problems: Single versus multi objective approaches. *Omega*, *92*, 102146. https://doi.org/10.1016/j.omega.2019.102146.

Daskalaki, S., Birbas, T., & Housos, E. (2004). An integer programming formulation for a case study in university timetabling. *Eur. J. Operational Res.*, *153*(1), 117-135. https://doi.org/10.1016/S0377-2217(03)00103-6.

Di Gaspero, L. & Schaerf, A. (2006). Neighborhood portfolio approach for local search applied to timetabling problems. *J. Math. Modell. Algorithms*, *5*, 65-89. https://doi.org/10.1007/s10852-005-9032-z.

Dimopoulou, M. & Miliotis, P. (2001). Implementation of a university course and examination timetabling system. *Eur. J. Operational Res.*, *130*(1), 202-213. https://doi.org/10.1016/S0377-2217(00)00052-7.

Domenech, B. & Lusa, A. (2016). A MILP model for the teacher assignment problem considering teachers' preferences. *Eur. J. Operational Res.*, *249*(3), 1153-1160. https://doi.org/10.1016/j.ejor.2015.08.057.

Faudzi, S., Abdul-Rahman, S., & Abd Rahman, R. (2018). An assignment problem and its application in education domain: A review and potential path. *Adv. Oper. Res.*, *2018*(1), 8958393. https://doi.org/10.1155/2018/8958393.

Herres, B. & Schmitz, H. (2021). Decomposition of university course timetabling. *Ann. Oper. Res.*, *302*(2), 405-423. https://doi.org/10.1007/s10479-019-03382-0.

Lee, S. M. & Schniederjans, M. J. (1983). A multicriteria assignment problem: A goal programming approach. *Interfaces*, *13*(4), 75-81. https://doi.org/10.1287/inte.13.4.75.

Lemos, A., Melo, F. S., Monteiro, P. T., & Lynce, I. (2019). Room usage optimization in timetabling: A case study at Universidade de Lisboa. *Oper. Res. Perspect.*, *6*, 100092. https://doi.org/10.1016/j.orp.2018.100092.

Lü, Z. & Hao, J. K. (2010). Adaptive tabu search for course timetabling. *Eur. J. Operational Res.*, *200*(1), 235-244. https://doi.org/10.1016/j.ejor.2008.12.007.

Mallari, C. B., San Juan, J. L., & Li, R. (2023). The university coursework timetabling problem: An optimization approach to synchronizing course calendars. *Comput. Ind. Eng.*, *184*, 109561. https://doi.org/10.1016/j.cie.2023.109561.

MirHassani, S. A. (2006). A computational approach to enhancing course timetabling with integer programming. *Appl. Math. Comput.*, *175*(1), 814-822. https://doi.org/10.1016/j.amc.2005.07.039.

Mokhtari, M., Vaziri Sarashk, M., Asadpour, M., Saeidi, N., & Boyer, O. (2021). Developing a model for the university course timetabling problem: A case study. *Complexity*, *2021*(1), 9940866. https://doi.org/10.1155/2021/9940866.

Müller, T., Rudová, H., & Müllerová, Z. (2024). Real-world university course timetabling at the International Timetabling Competition 2019. *J. Scheduling*, 1-21. https://doi.org/10.1007/s10951-023-00801-w.

Murray, K., Müller, T., & Rudová, H. (2007). Modeling and solution of a complex university course timetabling problem. In *Practice and Theory of Automated Timetabling VI: 6th International Conference, PATAT 2006* (pp. 189-209). Berlin: Springer. https://doi.org/10.1007/978-3-540-77345-0_13.

Nossack, J. (2022). Therapy scheduling and therapy planning at hospitals. *Omega*, *109*, 102594. https://doi.org/10.1016/j.omega.2022.102594.

Oladejo, N. K., Abolarinwa, A., Salawu, S. O., Bamiro, O. M., Lukman, A. F., & Bukari, H. I. (2019). Application of optimization principles in classroom allocation using linear programming. *Int. J. Mech. Eng. Technol.*, *10*(1), 874-885. https://eprints.lmu.edu.ng/id/eprint/2145

Phillips, A. E., Walker, C. G., Ehrgott, M., & Ryan, D. M. (2017). Integer programming for minimal perturbation problems in university course timetabling. *Ann. Oper. Res.*, *252*, 283-304. https://doi.org/10.1007/s10479-015-2094-z.

Phillips, A. E., Waterer, H., Ehrgott, M., & Ryan, D. M. (2015). Integer programming methods for large-scale practical classroom assignment problems. *Comput. Oper. Res.*, *53*, 42-53. https://doi.org/10.1016/j.cor.2014.07.012.

Song, T., Liu, S., Tang, X., Peng, X., & Chen, M. (2018). An iterated local search algorithm for the university

course timetabling problem. *Appl. Soft Comput.*, *68*, 597-608. https://doi.org/10.1016/j.asoc.2018.04.034.

Soria-Alcaraz, J. A., Özcan, E., Swan, J., Kendall, G., & Carpio, M. (2016). Iterated local search using an add and delete hyper-heuristic for university course timetabling. *Appl. Soft Comput.*, *40*, 581-593. https://doi.org/10.1016/j.asoc.2015.11.043.

Tan, J. S., Goh, S. L., Kendall, G., & Sabar, N. R. (2021). A survey of the state-of-the-art of optimisation methodologies in school timetabling problems. *Expert Syst. Appl.*, *165*, 113943. https://doi.org/10.1016/j.eswa.2020.113943.

Valouxis, C. & Housos, E. (2002). Combined bus and driver scheduling. *Comput. Oper. Res.*, *29*(3), 243-259. https://doi.org/10.1016/S0305-0548(00)00067-8.

Yinusa, A. & Faezipour, M. (2023). Optimizing healthcare delivery: A model for staffing, patient assignment, and resource allocation. *Appl. Syst. Innov.*, *6*(5), 78. https://doi.org/10.3390/asi6050078.

## Nomenclature

| | |
|---|---|
| $\Psi$ | The set of all curriculums. |
| $All\_courses$ | The set of all courses. |
| $All\_t\_slots$ | The set of all time slots, i.e., all pairs of weekdays and timeframes. |
| $All\_teachers$ | The set of all teaching staff in the school. |
| $courses(\Psi)$ | The set of all courses included in curriculum $\Psi$. |
| $T(\Psi)$ | The set of all time slots in which courses included in curriculum $\Psi$ can be taught. |
| $C(i)$ | The set of courses that lecturer $i$ is authorized to teach. |
| $N_j$ | The number of sessions to be offered for course $j$. |
| $h_j$ | Teaching load of course $j$. |
| $Min\_position_i$ | Minimal teaching load (in hours) of lecturer $i$. |
| $Max\_position_i$ | Maximal teaching load (in hours) of teacher $i$. |

## Appendix

**Table A1.** Courses – 2023 winter semester ("Tut" stands for "tutorial")

| Program | Course Symbol |
|---|---|
| EA_M_1 | EA_M-A1, EA_M-A1-tut[1], EA_M-A2, EA_M-A2-tut[1], EA_M-A3, EA_M-A3-tut[1], EA_M-A4, EA_M-A5, EA_M-A5-tut[1], EA_M-A6, EA_M-A6-tut[1] |
| EA_M_2 | EA_M-B1, EA_M-B1-tut, EA_M-B2, EA_M-B2-tut, EA_M-B3, EA_M-B4-tut, EA_M-B4, EA_M-B5, EA_M-B5-tut, EA_M-B6, EA_M-B6-tut |
| EA_M_3 | EA_M-C1, EA_M-C2, EA_M-C2-tut, EA_M-C3, EA_M-C4, EA_M-C5, EA_M-C6, EA_M-C6-tut, EA_M-C7, EA_M-C7-tut |
| EA_E-1 | EA_E-A1, EA_E-A1-tut, EA_E-A2, EA_E-A2-tut, EA_E-A3, EA_E-A3-tut, EA_E-A4, EA_E-A5, EA_E-A6-tut |
| EA_E-2 | EA_E-B1, EA_E-B1-tut , EA_E-B2, EA_E-B2-tut, EA_E-B3, EA_E-B3-tut, EA_E-B4, EA_E-B4-tut |
| EA_E-3 | EA_E-C1, EA_E-C2, EA_E-C2-tut, EA_E-C3, EA_E-C4, EA_E-C4-tut, EA_E-C5, EA_E-C6, EA_E-C7 |
| EA_E-4 | EA_E-D1, EA_E-D1-tut, EA_E-D2, EA_E-D3, EA_E-D3-tut |
| EM_M-1 | EM_M-A1, EM_M-A1-tut[1], EM_M-A2, EM_M-A3, EM_M-A4, EM_M-A4-tut[1], EM_M-A5, EM_M-A6, EM_M-A6-tut[1], EM_M-A7, EM_M-A7-tut[1] |
| EM_M-2 | EM_M-B1, EM_M-B2, EM_M-B2-tut[1], EM_M-B3, EM_M-B3-tut[1], EM_M-B4, EM_M-B4-tut[1], EM_M-B5, EM_M-B6, EM_M-B6-tut[1], EM_M-B7 |
| EM_M-3 | EM_M-C1, EM_M-C2, EM_M-C3 |
| EM_E-1 | EM_E-A1, EM_E-A1-tut[1], EM_E-A2, EM_E-A3, EM_E-A3-tut[1], EM_E-A4, EM_E-A4-tut[1], EM_E-A5, EM_E-A5-tut[1] |
| EM_E-2 | EM_E-B1, EM_E-B1-tut[1], EM_E-B2, EM_E-B2-tut[1], EM_E-B3, EM_E-B3-tut[1], EM_E-B4 |
| EM_E-3 | EM_E-C1, EM_E-C2, EM_E-C3 |
| ML_E-1 | ML_E-A1, ML_E-A2, ML_E-A3, ML_E-A3-tut, ML_E-A4, ML_E-A4-tut |
| ML_E-2 | ML_E-B1, ML_E-B2, ML_E-B3, ML_E-B3-tut, ML_E-B4, ML_E-B4-tut, ML_E-B5, ML_E-B6 |
| ML_E-3 | ML_E-C1, ML_E-C2, ML_E-C3, ML_E-C4, ML_E-C5, ML_E-C6, ML_E-C7, |
| M_M-1 | M_M-A1, M_M-A2, M_M-A2-tut[1], M_M-A3, M_M-A3-tut[1], M_M-A4, M_M-A4-tut[1], M_M-A5, M_M-A5-tut[1], M_M-A6 |
| M_M-2 | M_M-B1, M_M-B1-tut[1], M_M-B2, M_M-B3, M_M-B4, M_M-B5, M_M-B6, M_M-B7 |
| M_M-3 | M_M-C1, M_M-C2, M_M-C3, M_M-C4, M_M-C5, M_M-C6, M_M-C7 |
| M-1 | M-A1, M-A1-tut[1], M-A2, M-A3, M-A3-tut[1], M-A4, M-A4-tut[1], M-A5, M-A5-tut[1] |
| M-2 | M-B1, M-B1-tut[1], M-B2, M-B3, M-B4, M-B5, M-B6, M-B7, M-B8 |

| | |
|---|---|
| M-3 | M-C1, M-C2, M-C3, M-C4, M-C5, M-C6, M-C7, M-C8 |
| M_E-1 | M_E-A1, M_E-A1-tut[(1)], M_E-A2, M_E-A2-tut[(1)], M_E-A3, M_E-A3-tut[(1)], M_E-A4, M_E-A4-tut[(1)] |
| M_E-2 | M_E-B1, M_E-B1-tut[(1)], M_E-B2, M_E-B3, M_E-B4, M_E-B5, M_E-B6 |
| M_E-3 | M_E-C1, M_E-C2, M_E-C3, M_E-C4, M_E-C5, M_E-C6, M_E-C7 |
| | (1) 2 classes required for this tutorial. |

**Table A2.** Lecturers' eligible courses (FT – Faculty, T – Non-faculty)

| Lecturer | Eligible Courses |
|---|---|
| FT1 | M_E-A3, M_M-A4, M-A4, M_M-A5, M-A5, M-A5, EM_E-A2, EM_M-A3, EM_E-A4, EA_E-A5, EM_M-A6 |
| FT2 | EA_E-D2, EM_E-C3, M_M-C6, M-C7, M_E-C5 |
| FT3 | EA_M-C2, EA_E-D1, EA_M-A3, EA_E-A3, EM_E-A3, EM_M-A4 |
| FT4 | EA_E-D2, EM_E-C3, EM_M-C3, EA_M-A3, EA_E-A3, EM_E-A3, EM_M-A4 |
| FT5 | ML_E-C3, M_M-B4, M-B5, EM_M-B5, M_E-B2, M-B2, M_M-B2 |
| FT6 | M_E-C6, M_M-B5, M-B6, M_E-B5 |
| FT7 | EM_M-A7, EM_E-A5, EA_E-B1, ML_E-B4, M_M-A5, M-A5, M_E-A4 |
| FT8 | M-A2, EM_M-B7, EM_E-B4, M_M-B7, M-B8, M_E-B6 |
| FT9 | EA_E-C7, EA_E-D2, EA_E-C5, EM_M-C2, EM_E-C2, M_M-B3, M-B3 |
| FT10 | EA_M-C4, EA_E-C6 |
| FT11 | ML_E-B3, EM_M--B4, ML_E-A2, ML_E-C2, ML_E-B5 |
| FT12 | EA_E-A4, M_M-B3, M-B3, EA_E-C5, EM_M-C2, EM_E-C2 |
| FT13 | M_M-B1, M-B1, M_E-B1, EA_M-B4, EA_E-B2 |
| FT14 | EA_M-A1-tut, EA_M-B1, EA_M-C6, EA_E-C4 |
| FT15 | ML_E-B1, ML_E-C6 |
| FT16 | M-A2, M_M-C4, M-C5 |
| FT17 | EA_M-B3, EA_E-C1, EA_E-D3-tut, EA_E-C4, EA_M-B2, EA_E-D3 |
| FT18 | M-B3, EA_E-C5, EM_M-C2, EM_E-C2, EA_E-D2, EA_E-C7 |
| FT19 | EM_E-A2, EM_M-A3, EM_E-A4, EA_M-A5, EA_E-A5, EM_M-A6, ML_E-A4, M_E-A3, M_M-A4, M-A4 |
| FT20 | EA_E-B4-tut, EM_M-B6-tut, EM_E-B3-tut, |
| FT21 | ML_E-C4, ML_E-A3, M_M-A3, M-A3, M_E-A2 |
| FT22 | M-C3, M_E-C3, ML_E-C3, M_M-B4, M-B5, M-C5 |
| FT23 | M_M-A3, M-A3, M_E-A2, EM_M-B3, EM_E-B2, EA_M-B5 |
| FT24 | ML_E-C6 |
| FT25 | M-C2, M_E-C2, ML_E-C1, EM_M-C1, EM_E-C1 |
| T1 | EA_E-A1-tut, EM_E-A1-tut, M_M-A2-tut, M-A1-tut, EM_M-A1-tut |
| T2 | M-B1, M_E-B1, M_M-B1-tut, M-B1-tut, M_E-B1-tut |
| T3 | M-A2 |
| T4 | M_E-A2, M-A3, M_M-A3-tut, M-A3-tut, M_E-A2-tut, EM_E-B2-tut |
| T5 | M_M-B7, M-B8, M_E-B6, EM_M-B7, EM_E-B4 |
| T6 | M-A3-tut, M_E-A2-tut, M_M-A3-tut, EA_M-A6-tut, M_M-A5-tut, M-A5-tut, M_E-A4-tut, ML_E-B4-tut, ML_E-B3-tut, EM_M-B4-tut |
| T7 | M-C4, M_E-C7 |
| T8 | M_M-A4-tut, M-A4-tut, M_E-A3-tut, EM_E-A4-tut, EA_E-A6-tut, EA_M-A5-tut |
| T9 | EM_M-A7-tut, EM_E-A5-tut, EA_E-B1-tut, ML_E-B3-tut, EM_M-B4-tut |
| T10 | M_M-B4, M-B5 |
| T11 | EM_M-A5 |
| T12 | M-C5, M_M-C6, M-C7, M_E-C5 |
| T13 | M-A4-tut, M_E-A3-tut, EM_E-A4-tut, EA_E-A6-tut, EA_M-A5-tut, ML_E-A4-tut, EM_M-A6-tut |
| T14 | EA_M-C5 |
| T15 | M_M-C1, M-C1, M_E-C1 |
| T16 | M-B6, M_E-B5, M_M-C6, M-C7, M_E-C5 |
| T17 | M_M-B1, M-B1, M_E-B1, EA_M-B4, EA_E-B2 |
| T18 | M_M-C7, M-C8, M_E-C6, M_E-C7, M_M-B6, M-B7 |
| T19 | M-B2, M_E-B2, ML_E-A1 |
| T20 | EA_E-D1-tut, EA_M-B3, EA_E-C1 |
| T21 | EA_E-A1, EM_E-A1, M_M-A2, M-A1, EM_M-A1 |
| T22 | EA_E-A2, EM_E-B1, M_E-A1, EM_M-B2 |
| T23 | M-B3, EA_E-C5, EM_M-C2, EM_E-C2 |
| T24 | M-B4, M_E-B3, EM_M-A2 |
| T25 | EA_E-C3, EA_M-C7 |
| T26 | EA_E-C2, EA_E-C2-tut |
| T27 | EA_E-B4, EM_M-B6, EM_E-B3 |
| T28 | EA_M-C6-tut, EA_E-C4-tut |
| T29 | EA_M-A5-tut |

| | |
|---|---|
| T30 | EA_E-B2-tut, M_M-B1-tut, M-B1-tut, M_E-B1-tut |
| T31 | EA_E-B2-tut, M_M-B1-tut, M-B1-tut, M_E-B1-tut |
| T32 | EA_E-A1-tut, EM_E-A1-tut, M_M-A2-tut, M-A1-tut, EM_M-A1-tut |
| T33 | EA_E-A2-tut, EM_E-B1-tut, M_E-A1-tut, EM_M-B2-tut |
| T34 | EA_E-A2-tut, EM_E-B1-tut, M_E-A1-tut, EM_M-B2-tut |
| T35 | EA_M-A3-tut, EA_E-A3-tut, EM_E-A3-tut, EM_M-A4-tut |
| T36 | EA_M-A3-tut, EA_E-A3-tut, EM_E-A3-tut, EM_M-A4-tut |
| T37 | EA_E-B3-tut, EM_M-B3-tut, EM_E-B2-tut, EA_M-B5-tut |
| T38 | EA_E-B3-tut, EM_M-B3-tut, EM_E-B2-tut, EA_M-B5-tut |
| T39 | M-A5-tut, M_E-A4-tut |
| T40 | M-A5-tut, M_E-A4-tut |
| T41 | EM_M-A7-tut, EM_E-A5-tut, EA_E-B1-tut |
| T42 | EA_M-A1-tut, EA_E-D3-tut, EA_M-C1, EA_E-D3 |
| T43 | EM_M-A7-tut, EM_E-A5-tut, EA_E-B1-tut |
| T44 | EM_M-B5, EA_M-B3, EA_E-C1 |
| T45 | M_M-C7, M-C8, M_M-B5, M-B6, M_E-B5 |
| T46 | EA_E-C7, EA_E-D2 |
| T47 | EA_M-C3, EA_M-C4, EA_M-C5 |
| T48 | M_E-B4, M_M-C6, M-C7, M_E-C5 |
| T49 | M_M-A3-tut, M-A3-tut, M_E-A2-tut |
| T50 | M_M-C5, M-C6, M_E-C4, M-B6 |
| T51 | M_M-C5, M-C6, M_E-C4 |
| T52 | M_M-B5, M-B6, M_E-B5, M_M-B6, M-B7 |
| T53 | EA_M-A5-tut |
| T54 | EA_E-B4-tut, EM_M-B6-tut, EM_E-B3-tut |
| T55 | M_M-A3, M-A3, M_E-A2, M_M-A5, M-A5, M_E-A4, EM_M-B1 |
| T56 | EA_E-A1, EM_E-A1, M_M-A2, M-A1 |
| T57 | ML_E-A2, M_M-A5, M-A5, M_E-A4 |
| T58 | EM_M-A5 |
| T59 | EA_E-D3 |
| T60 | EA_E-C6 |
| T61 | M_M-B5, M-B6, M_E-B5, M_M-C6, M-C7, M_E-C5 |
| T62 | EM_M-A7, EM_E-A5, EA_E-B1 |
| T63 | EM_M-A7, EM_E-A5, EA_E-B1 |
| T64 | EA_E-D1, EA_M-C2-tut, EA_E-D1-tut |
| T65 | EA_M-A3-tut, M_M-A3-tut, M-A3-tut, M_E-A2-tut, EM_E-B2-tut |
| T66 | EA_M-B5-tut |
| T67 | ML_E-B3-tut, EM_M-B4-tut |
| T68 | EA_E-B3-tut, EM_M-B3-tut, EM_E-B2-tut, EA_M-B5-tut |
| T69 | M-A5-tut, M_E-A4-tut |
| T70 | M-A5-tut, M_E-A4-tut |
| T71 | M_E-C5 |
| T72 | M-A2 |
| T73 | EA_M-B2-tut |
| T74 | EM_M-A6 |
| T75 | M_M-A5, M-A5, M_E-A4 |