A REPORT OF SORTS

For schools, colleges AND other institutions in general with an operational structure akin to a teacher (HANDLER)- class - ARM PARADIGM.

Oluwatomilayo Inioluwa, OWOEYE

09012481174

Oluwatomilayoowoeye.oio@gmail.com

2022

Table of Contents

[THE TIMETABLE PROBLEM 2](#_Toc110933171)

[THE MEANING OF TERMS AS USED IN THIS APP 3](#_Toc110933172)

[THE APP IN THE RUNNING 5](#_Toc110933173)

[MODEL HIERARCHIES 5](#_Toc110933174)

[DYNAMICS OF THE TEACHER MODEL 6](#_Toc110933175)

[THE SORTING PATTERN: TERMS AND CONCEPTS 7](#_Toc110933176)

[THE ATPG SCALE 7](#_Toc110933177)

[THE CHUNK VALUE 7](#_Toc110933178)

[PACKETING 8](#_Toc110933179)

[PACKETING PATTERNS 8](#_Toc110933180)

## THE TIMETABLE PROBLEM

The rigor of fashioning a timetable for a large school with so many areas of compartmentalization could be, among other things, overwhelming. Teachers usually take on more than one subject across more than one class and are required to fulfil the weekly quota of each subject they take for each of the classes they take. The timetable for such a system is thus constructed such that for all the periods available for each arm of each class for a single day, each teacher’s teaching periods do not coincide or overlap for every subject and every class that they teach, so they would not be required to be in more than one place at a time, which would be grossly impractical – if not impossible.

Setting up the timetable, more often than not, becomes very complicated *very* quickly as other requirements (*double* periods being one of many) often get thrown into the mix. The larger the school, the larger the teaching staff to pass subjects around, or the larger the number of subjects, the more complicated the problem tends to prove, especially when adjustments are required on the fly.

The launch of such a task presents itself as hectic, and would require a lot of permutation.

Permutations, it turns out, is where computers shine the brightest.

This project solves the timetable problem by creating a software package that abstracts away the hassle of fashioning out a timetable, i.e. sorting the teachers and their classes into periods spread across each day for each day spread across each week such that:

1. no two (or more) teachers are assigned the same period for the same set of students on the same day.
2. no two (or more) exact periods in different class arms are assigned to the same teacher.
3. every subject’s weekly period quotas (frequencies) are met.
4. students’ convenience, subject structure and Teachers’ expertise are put squarely into perspective.

Many more requirements that take into consideration the efficiency of teaching (for the teacher) and learning optimization (for the students) have been factored right into the algorithm driving the code.

## THE MEANING OF TERMS AS USED IN THIS APP

The purpose of this section is to get the reader conversant with the terms and processes involved in the use of this app. Indeed, these processes are not complicated, they just might require a little bit of getting used to, as some of the terms (with which we probably have already been familiar) have been redefined. The redefined terms have undergone no radical change in definition, rest assured; they just have been tweaked in a few places for the purpose of the app. For the most part, they retain their conventional, outside-this-app meanings to a significant degree.

There exists the likelihood that some of the terms used in this application might be a little confusing to new users. These terms are thus defined:

1. **Day (as in School day).**

Day, as used in this app, does not necessarily refer to its dawn-till-dusk meaning. It simply refers to the time frame (usually a day, hence the name) within which all the periods for a certain class feature. It spans the first and last periods of school for an (actual) entire day in real-life. For a typical school in South-Western Nigeria (where this app was initially written), the school day starts by 8:00 a.m. and runs till 2:00 p.m. unless there are arrangements for afternoon classes. It is the first period all the way till the last.

1. **Department:**

The Department, much like its real-life meaning, is a container for the subjects provided by a school to its students. It is a catalogue for organizing subjects which share some level of a decided-upon similarity.

1. **Subject/Course**

Subject (also, Course), as with "Day" above, may not always refer to the conventional meaning of the word in an academic setting, though it does, more often than not. A subject (course), with regard to the timetable, is any engagement which occupies one or more periods of a given class arm. Subjects occupy whole number values of periods e.g. Music in the first period or Math in the first two periods. Two subjects can only exist for the same period(s) if they go concurrently and both run through till the end of the period, e.g. Yoruba/Igbo languages occupy the last period. Subjects do not always have to be academic; break (recess) or extracurriculars also count as subjects inasmuch as they fall within "Day". A subject is termed academic if it is handled by a teacher. Non-academic periods are termed "Special periods".

1. **Period:**

A period is the set duration within which a subject occurs. How long it takes is institution-specified. It is the very measure by which "Day" is calibrated. Academic periods tend to have the same period duration, whereas, non-academic periods (e.g. Break or Recess) might take varying lengths.

1. **School Class Category:**

Also termed "class category" or "class group", this is simply a collection of all the classes across different levels or grades with a common property for ease of reference. This in real life is what we would call "Senior School", or "Junior High" in a school situation especially where more than one school class category might exist. For instance, the typical Nigerian secondary school has two class categories: Junior secondary school and Senior secondary school.

1. **School Class (or simply, Class):**

The school class is a combination of all the arms that share one level of the academic hierarchy in the school setting. In the typical Nigerian school system, its arms are prefixed with the its name and alphabetized, e.g. JSS 1A, SSS 2B. This application is able to render class arms in said format. Also, each arm of the class in question can be rendered numerically, e.g. JSS 1\_1, SSS2\_2.

1. **School Class Arm:**

Sometimes, plainly referred to as "Arm", the school class arm is a sub-section of the school class, which in turn is a subset of the school class category. The arm is the compartment into which the school class divided, primarily to accommodate a number of students that exceeds the standard number of students studying together in the same class room and to allow for different specializations between students of the same class, e.g. Sciences and The Arts. The school class arm is the literal classroom in which schoolwork is done.

1. **Teacher**:

Fairly self-explanatory. A teacher handles the subject for a class at a given "Period" of the "Day". It is quite possible for a teacher to take on more than one subject, across more than one class arm.

1. **Teacher’s Sequence**

The teacher’s sequence is the list of class arms for which the teacher is responsible to their subject.

## THE APP IN THE RUNNING

### MODEL HIERARCHIES

This application models the operational structure of a full-fledged (typical Nigerian) school. However, this model structure is in no way limited to schools with such an operational structure. It can be modified to suit schools whose structures are not as involved. As earlier stated, this application models both the school's operational structure and the (likely) interactions between them. These models present themselves in hierarchical formats.

There are two models, conveniently termed The Class-hierarchy and The Course-hierarchy. At its core, the timetable, in this context, is little more than a black and white representation of the relationship between subjects (courses) and classrooms at certain parts of the day, both collectively (i.e. in whatever groups these might have been sorted into) and individually. Both models would have to be created.

#### The Class-hierarchy

The class hierarchy represents the superset-set-subset relationship of the between the items in the class hierarchy.

The hierarchy can be simply represented as:

Class group > Class > Class arm

Class groups contain all the classes which in turn contain all the arms for said classes. The class category in the real-life school setting is what we would call 'Junior School' or 'Senior High' or 'middle School' (US). It is merely a convenient way to categorize classes, especially in a setting that has more than one class category running.

#### The Course hierarchy

The Course hierarchy, just like the class hierarchy above, is a superset-set-subset relationship that describes the key components of the structure that handles and executes teaching, course structure and the various instances of specializations of class arms within a class, e.g. class arms designated for The Arts or Sciences or other.

The teachers primarily handle the teaching and what they teach is the subject or course. The department is merely the container that houses like subjects, a neat way of grouping subjects in the event that similar-structured subjects are taken as a whole. However, for ease of design (as regards the efficiency of coding), among other things, the subject is said to "have" a teacher and "hail from" a department.

It should be plainly stated that there are certain subjects (special subjects, as they are referred to here, as discussed in the definition of terms section) which do not require handling by a teacher. These are subjects like break-time (recess) or extracurricular activities. (The definition of the term “subject” used here and throughout the manual is consistent with the definition – or redefinition, if you will – in the Definition of terms section).

The Course hierarchy goes:

Department > Subject/Course > Teacher

Except in the case of a **special subject** where the subject stands alone, without a department and teacher.

### DYNAMICS OF THE TEACHER MODEL

Apparently, the Teacher model (as is in the app), is an "in binary" rendering of the teacher, who, much more often than not, is an actual human. There exist some intricacies with regard to the teacher model that can neither be ignored nor compromised; not even a little.

Some immediately come to mind.

1. **There should be no intersection between any of the periods in the teacher’s daily sequence.**

The teacher’s periods for a particular day must not overlap.

This is a major constraint in the setup of any timetable. Although, teachers (might) teach across different class arms, probably across different classes, for the day in which they teach, there should be no intersection (or clash) whatsoever between periods in the teacher's sequence. Otherwise, that would imply that the teacher is required to be in more than one class arm at a time.

1. **Some of the teaching staff might be required to be "off work " on certain days of the week.**

This is especially true for the National Youth Service Corps Members (A Nigerian one-year assignment programme for tertiary institution graduates, especially for those under thirty) who during their term of service often teach in schools. They are required by the programme to gather on a certain day to effect their community-development strategies, and as such do not attend school on that day. Indeed, this is just one use-case, nevertheless, the possibility that teachers would, for some legitimate reason, be consistently absent on a certain day or days of the week is plausible.

This application puts these two factors squarely into perspective and would fulfill these requirements for every teacher as much as it can, even if recalibrations and readjustments would have to be carried out.

## THE SORTING PATTERN: TERMS AND CONCEPTS

### THE ATPG SCALE

**The need for an objective measure of sorting subjects with regard to which comes first in the list of periods of a class arm for a particular day.**

ATPG stands for **Arithmetic, Theoretical, Practical, Grammatical**. This is a rating system that evaluates subjects based on these four structural qualities by assigning said qualities numerical weights and then multiplying said weight by their rating value, which is given in the graphical user interface of the app and runs from 1 through 10). No subject scores 0 in this rating because every academic subject has a measure of all these qualities, however little.

The different weights attached to different qualities already ensures that different courses/subjects would score differently according to the ATPG scale; some higher, some lower.

Where:

weight of the “Analytical” property of the subject/course

weight of the “Theoretical” property of the subject/course

weight of the “Practical” property of the subject/course

weight of the “Grammatical” property of the subject/course

rating of the “Analytical” property of the subject by the user from the user interface.

rating of the “Theoretical” property of the subject by the user from the user interface.

rating of the “Practical” property of the subject by the user from the user interface.

rating of the “Grammatical” property of the subject by the user from the user interface.

is simply a scaling constant.

The subjects are then sorted according to their ATPG values in descending order. This sort is maintained throughout except in cases where an overlap would occur in a teacher’s periods.

**N.B.**

The score gotten by a subject in the ATPG analysis is in no way a measure of the importance of said subject in reality, whether singly or with respect to the others. The ATPG analysis merely yields a numerical value with which to sort subjects. Academic subjects, that is.

Non-academic (special) subjects have no ATPG value.

### THE CHUNK VALUE

To begin with, every subject has its weekly frequency for every class arm that takes it; for instance, Drama could take up 4 periods weekly for JSS 1A. Also, every time it features (on the day in which it features), it could take up two consecutive periods at a time (double-period). Thus, it only features on 2 days of the week, for 2 consecutive periods on each of these days.

The number of (consecutive) periods the subject takes for each feature is called its *chunk value*, or simply, its *chunk.*

The chunk value of each subject is specified by the user right from the user interface.

### PACKETING

Packeting refers to the process of dishing out subjects into different days of the week for a particular class arm. This “dishing out” is based on the chunk value of said subject. For instance, as in the scenario of *Drama* earlier stated, Drama would be assigned to say, Monday (twice, as its chunk value is 2) and on another day, say Thursday, for its remaining two periods, since (as earlier stated) its weekly frequency for JSS 1A is 4.

Also, the packeting process takes into account the maximum number of periods allowed for a class arm during the day(s) in question.

### PACKETING PATTERNS

Subjects, for each class arm, are not “packeted” into days of the week willy-nilly. The subjects are imputed into days of the week based on an array of mathematical patterns. These patterns make up what is referred to here as the *packeting algorithms*. Prefixed with a “D-” to represent days.

They are as follows:

1. **The D-Leapfrog packeting patterns:**

*The “Let’s quickly get this over with” pattern.*

One of the simplest of the algorithms, the D-Leapfrog packeting algorithm, much like the leapfrog game (whence came the name of this algorithm), has a lot to do with “skipping over” days of the week as it packets subjects.

Subjects are packeted into days, with one day in between any two neighbouring days in which the subject features. It follows the Monday-Wednesday or Tuesday-Thursday pattern.

Step-wisely, its operation is primarily this:

* It picks a subject to packet.
* Then it picks a day (say Monday).
* It checks if the maximum number of periods for the day have not been exceeded i.e. if there is vacancy.
* If vacancy exists and is large enough to fit its chunk value, it shoves the subject into the day according to its chunk value. If there is no vacancy, it simply traipses on to the next day and tries its luck.
* In the event that Monday works, (the leapfrog analogy plays in here), it “skips over” Tuesday and goes to Wednesday. And so on.
* If after reaching the end of the week like so, it still isn’t through packeting said subject, it sweeps back to days it skipped.

**Real-world implication**

This algorithm ensures that when subjects/courses feature, they do so close to each other during the week, only with a day in between. The nuances to this arrangement are largely perspective-dependent: the teacher’s and the student’s.

From the student’s point of view, the subject featuring on days, one within close proximity of the others means that the teacher’s class is still fresh in their memory and that they only have “worry” about a particular subject only for a small interval within the week, as opposed to when the subjects occupy both ends of the week.

From the teacher’s, it means that he (any gender identity applies) only has to actively concern himself with the preparation for teaching in a particular class arm for a close-packed sequence of days during the week, e.g. Monday through Wednesday or Tuesday through Thursday. His attending class for said class arm does not span the length of the entire week, except he has *that* many features. He teaches according to his chunk value on the days in which he teaches, i.e. single, double or – if you will – triple or other periods.

Altogether, the subject features (almost) back to back and serves the imperative of “Let’s (quickly) get the lecture on this subject over with”.

1. **The D-Center-cluster Pattern**

*The “move outward from the middle” pattern*

When the packeting process follows this pattern, subjects are first packeted into the day at the middle of the week, and as soon as the vacancy in said day (for subjects) runs out, it spreads to the days before and after it, on and on till it all the subjects spread evenly over the week.

This is (slightly) akin to pouring water in a bowl from a tap. Although, the water from the tap only hits a particular area of the bowl (the center, ideally), the water spreads till it is evenly distributed across the bowl.

The name “...Center-cluster” is because this pattern initially clusters all the subjects at the middle of the week before spreading them outward.

**Real-world implication**

**Write me**

1. **The DXLX-Reflection Algorithm**

*"The other as much from the end as the one is from the start."*

When subjects are packeted into days based on this pattern, said subjects are packeted into days spread out such that the days occupy both ends of the week first, before filling the middle of the week.

Any pair of days (into which the subjects are packeted) are situated such that how far the one day is from the start of the week is how far the other is from the end of the week. For a Monday-to-Friday week, examples of how pairs such as this would fill an entire week would be: Monday-Friday, Tuesday-Thursday and then Wednesday.

Thus, for the two days in a pair, if the first is at position x in a week of length of L, its counterpart (reflection) sits at the position L - x. That is the origin of the name "X-L-X".

**Real-world implication**

Sorting according to this algorithm means that subjects feature at the ends, as opposed to being back-to-back. This means that as soon as students are done taking the first bout of the subject, they need not concern themselves with having the lesson for the subject anytime soon. This arrangement could serve students well in the case of tests and quizzes where a back-to-back occurence of the class may not be all that desirable. Except of course, in the event that the students want the subject gotten over with as soon as possible.