## timeslot\_optimizer.py Optimization Tool User Manual

### Purpose:

This optimization tool assigns USC Marshall classes to appropriate timeslots. The optimization tool mainly looks to optimize for the following three criteria:

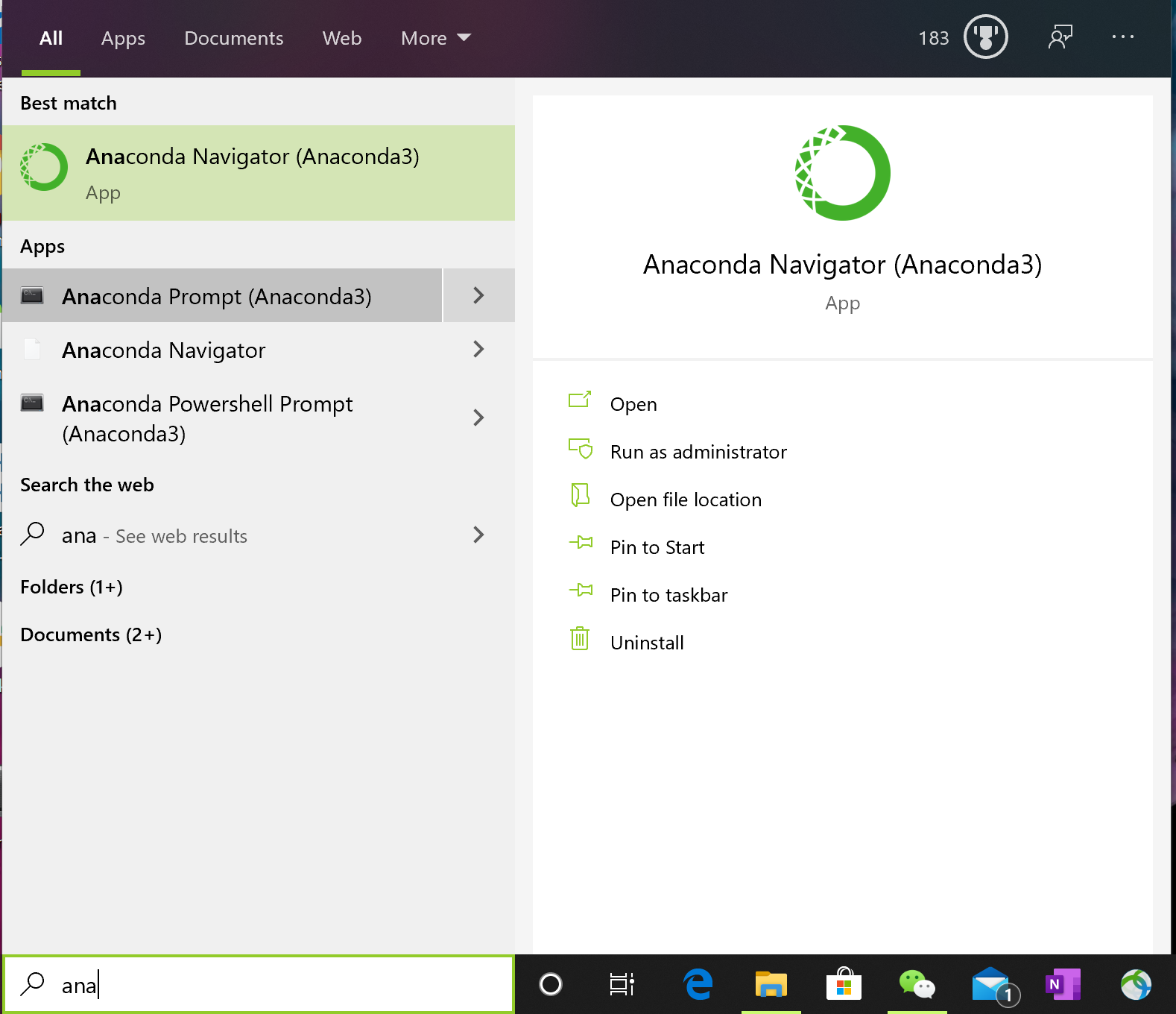
1. Minimize the cross-department differences in student and faculty satisfaction score (input from surveys), therefore making all the departments have similar satisfaction levels
2. Minimize the maximum total units assigned to any professors on any given day.
3. Minimize the units per day difference between weekdays. (Friday excluded because Mondays to Thursdays are often demanded by 4-unit classes and therefore should be busier.)

### Function Arguments:

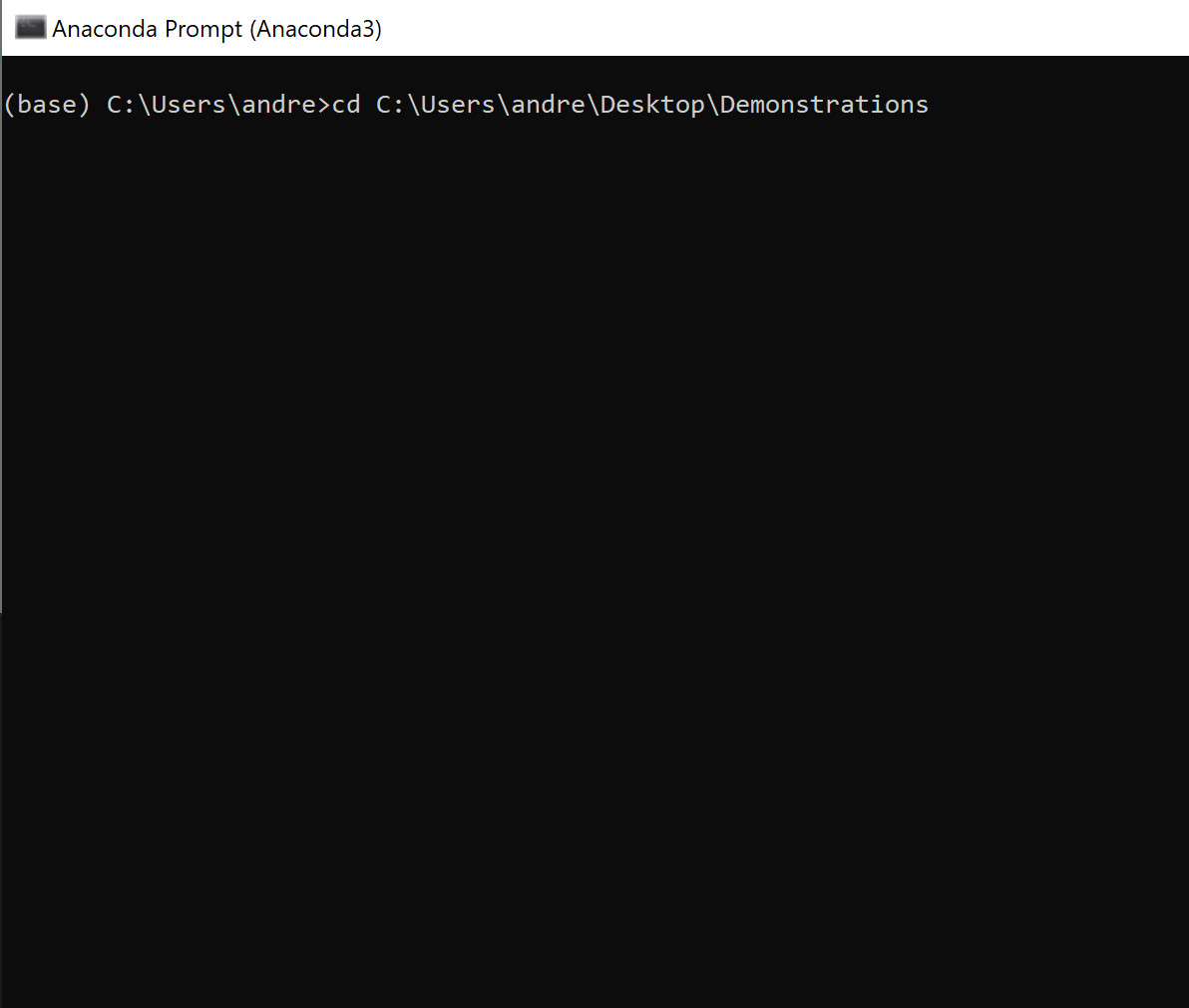
* **inputFile (str):** The input data file name. Please include the .xlsx extension for input excel file
* **outputFile (str):** The desired output data file name, also include .xlsx extension
* **classes\_same\_time (int):** (Optional) The maximum number of classes that can be assigned to one time slot (Default=60). Raise/ lower accordingly.
* **w\_z (float):** (Optional) The weight of faculty load factor in objective (Default=0.1). Raise this parameter if desired max units assigned to a professor allowed for any given day should be lowered.
* **w\_f (float):** (Optional) The weight of faculty importance factor in objective (Default=0.5). Raise this parameter if professor satisfaction is regarded more important.
* **w\_s (float):** (Optional) The weight of student importance factor in objective (Default=0.5). Raise this parameter if student satisfaction is regarded more important.
* **w\_w (float):** (Optional) The weight assigned to day of week balance in objective (Default=0.5). Raise this parameter if Monday to Thursday units assigned differences are considered more important.
* **install\_pandas (bool):** (Optional) Whether the user needs to install pandas to use this tool (Default=False). If pandas package is not installed on your system, please put True for this argument to install the package.

### Instructions:

1. Please make sure the input file and the timeslot\_optimizer.py file are in the same folder.
2. Open Anaconda Command Prompt by typing “Anaconda Command Prompt” in Windows search bar.



1. In Anaconda command prompt, type “cd /PATH/TO/YOUR/DIRECTORY” replace the path to your directory part with the directory where the input file and the timeslot\_optimizer.py file are located and press Enter.

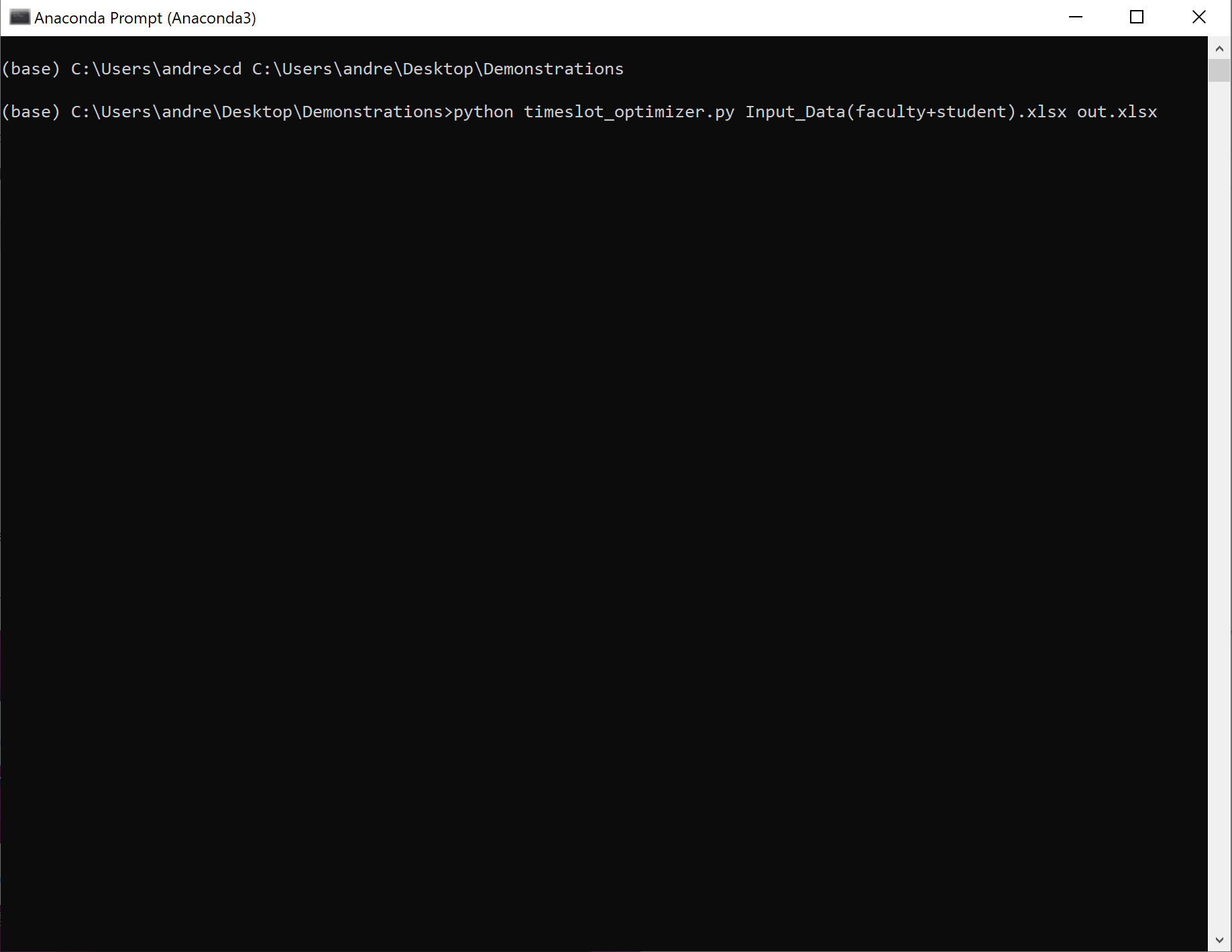


Example, the folder containing my timeslot\_optimizer.py and input files is : C:\Users\andre\Desktop\Demonstrations

1. If you are trying to generate a default set parameters and do not wish to change them during this optimization, please enter:

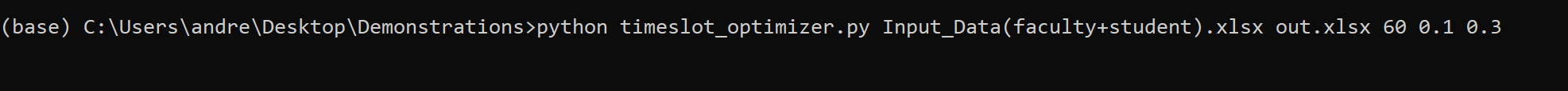
python timeslot\_optimizer INPUT\_FILE\_NAME.xlsx DESIRED\_OUTPUT\_FILE\_NAME.xlsx

and press enter. Replace the INPUT\_FILE\_NAME and DESIRED\_OUTPUT\_FILE\_NAME by your actual input file name and desired output file name. Note that all the parameters after “python” are separated by spaces, therefore please make sure the file names do not have spaces in them.



Example: the demonstration input file name was Input\_Data(faculty+student).xlsx and the desired output file name was out.xlsx.

1. If you do desire to change any of the default parameters, please enter them in order and please only enter numbers. If you only wish to change part of the default arguments, you still need to all the values all the way up to the position of that default argument you wish to change.

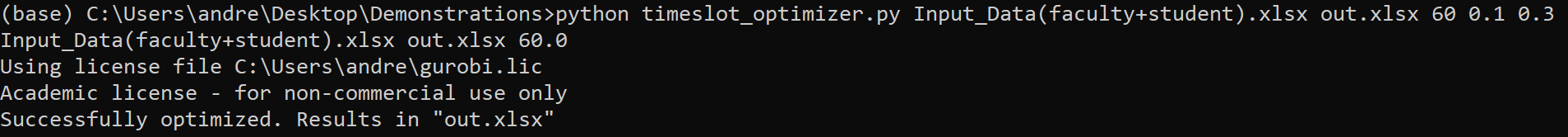


Example1 : if I wished to change the **w\_f** factor from 0.5 to 0.3, I need to enter all the default values all the way up to **w\_f.** (**classes\_same\_time** and **w\_z** are mandatory if I want to change **w\_f**).



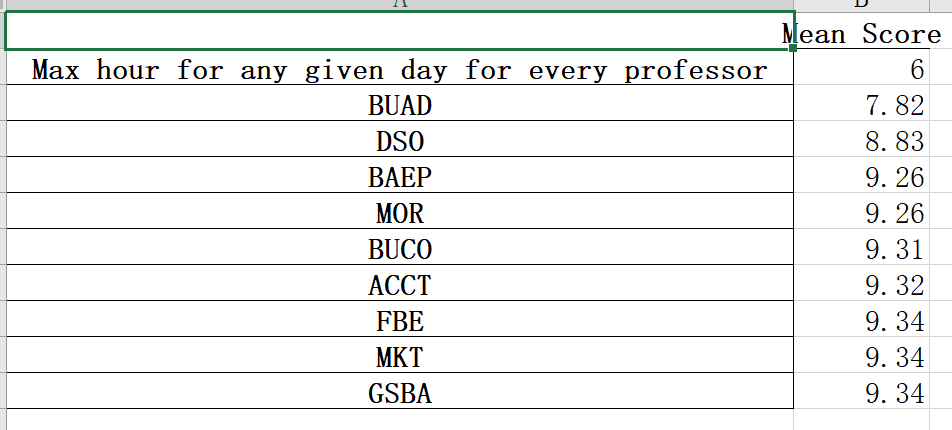
Example2: Therefore, if you wish to install pandas, you need to toggle **install\_pandas** to True, then you need to manually enter all default parameters. Refer to Function Arguments section for default values.

1. After hitting Enter, please wait for Gurobi to optimize and when you are greeted with this message “Successfully optimized. Results in x.xlsx“, then the optimization has finished. Please close command prompt and navigate to the folder containing the .py file and the input data .xlsx file. You will find a newly created .xlsx output file with the name you specified containing the optimization results from the optimizer.



### Output file:

Summary Sheet: The first sheet is the summary sheet for faculty and department after assigning units by optimization. The first row will contain the max units for any give professor in any day. Then the rest will contain mean satisfaction scores for each respective department after courses timeslot assignments. This can be a great place to check if the optimization is working and the gap of mean scores between departments should be minimized by our optimization tool.



Assigned Times: this sheet will contain all the courses and its respective units assigned times. Two units’ classes will have consecutive classes back to back, and four unit classes will have back to back and also occupy their Monday Wednesday or Tuesday Thursday pair. Note that

M = Monday

T = Tuesday

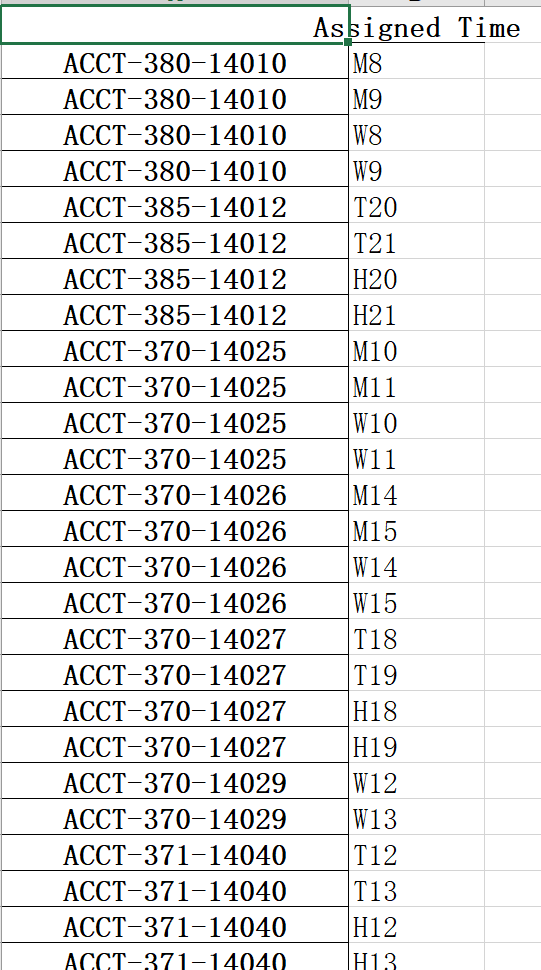
W = Wednesday

H = Thursday

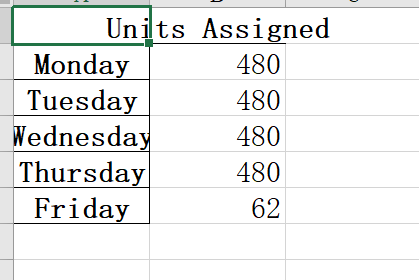
F = Friday

And the numbers behind the letter correspond to the start time for a given unit on a 24-hour clock.

This sheet is to clearly show the user of the output of our optimization tool. It has every class assigned to its respective timeslot.



Day of Week Assignment Stats: this sheet contains a summary statistic broken down by day of the week. This is also a good place to check if our objective is correct. Since we want the gap of units assigned to any given day to be as minimal as possible (except Friday). This should show that there is a minimal difference between Monday to Thursday unit assignments.



XXX Schedule: sheets 4 to 12 are the master time table for each of the nine departments in USC Marshall. The columns are the day of the week and the rows are the starting times of course units. Each cell contains a list of units that are assigned to the timeslot. These sheets can be then distributed to their respective departments for reference.

