

Assignment 3

When completed, save your workbook as a single Excel file (.xlsx extension) and submit through eClass.

Part 1: Creating a Mock Inventory

This part of the assignment teaches ways you can use built-in text and logic (IF) functions to create spreadsheets for various purposes. See Chapter 4 of the glade manual supplied with this document for a list and description of these functions.

Step 1: Worksheet Set-Up

- Download the accompanying file *sportingItems.txt*. It contains a list of data that can be imported into Excel. Open/create a new Excel document. Name the open worksheet **Mock Inventory**.
- Open *sportingItems.txt*, select all its data (ctrl/cmd + A) and copy (ctrl/cmd + C) it. Paste the data (ctrl/cmd + V) you copied onto *Mock Inventory* so that headings appear on Row 1.
- The inventory contains a list of 100+ sporting items with unique 4-digit IDs, the Sport they are associated with, respective dates (Day, Month, Year) they were acquired, and prices.
- Format column G to display *prices* in *Currency Format*. You may use any currency symbol other than \$. Make sure to use only one symbol throughout. Give headings **Category Code**, **Item Code**, **Stock Code**, **Stay (days)**, **Discount**, **Adjusted Price**, **Tax**, **Sale Price** to columns H to O.
- Name the data in columns A through O **id**, **sport**, **item**, **day**, **month**, **year**, **price**, **cat_code**, **item_code**, **stock_code**, **stay**, **discount**, **adjusted_price**, **tax**, **sale_price** respectively.
- Format columns L (Discount), M (Adjusted Price), N (Tax) and O (Sale Price) so that their data be displayed in percentages (%) and chosen currency respectively.

	A	B	C	D	E	F	G
1	ID	Sport	Item	Day	Month	Year	Price
2	5107	Golf	Bag	28	11	2020	\$199.99
3	4460	Golf	Cart	12	12	2020	\$1,899.00
4	5846	Golf	Shoe	13	2	2021	\$89.99
5	9922	Golf	Balls	21	11	2020	\$29.99
6	7793	Golf	Glove	13	2	2021	\$47.85
7	9819	Golf	GPS Watche	6	2	2021	\$350.97
8	3111	Golf	Bottom	30	1	2021	\$59.99
9	5753	Golf	Head covers	8	8	2020	\$28.89
10	1462	Golf	Outerwear	9	1	2021	\$100.00
11	7805	Golf	Top	23	1	2021	\$28.99
12	6411	Tenni	Racquet	30	1	2021	\$89.99
13	8705	Tennis	Shoe	10	10	2020	\$99.99
14	9882	Tennis	Bag	21	11	2020	\$79.99
15	9353	Tennis	String	15	8	2020	\$19.99
16	4679	Tennis	Balls	17	10	2020	\$17.99
17	1003	Tennis	Top	13	2	2021	\$49.99

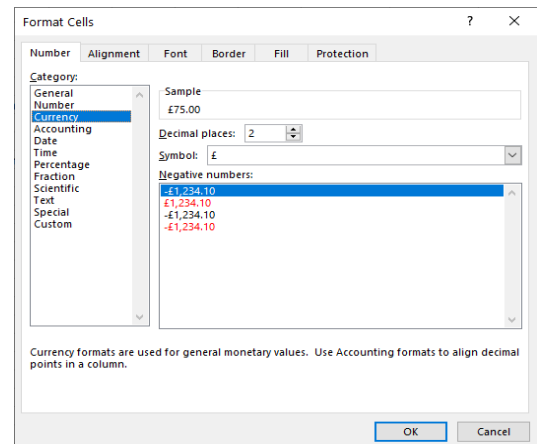


Figure 1: Prepared worksheet, Formatting data

Step 2: Formulas

- Write appropriate formulas using appropriate text functions to generate data for columns H through O. **Ensure that without any manipulation, formulas written should produce desired results for the given data and new data that may be added.** Function hints: IF(), LOWER(), UPPER(), FIND(), ISERROR(), CONCATENATE(), LEFT(), MID(), etc.
 - Column H (Category code):** Derive 4-letter category codes from the respective Sport categories in column B. If the name of the sport consists of one word, the code should be formed from the first four letters of that word. If the name of the sport consists of more than one word, form its code from the first three letters of the first word and the first letter of the second word. Category codes

should be in uppercase letters. For example, the code for Hiking should be *HIKI*, and that for Speed Skating (if added later) should be *SPES*.

- ii. **Column I (Item code):** Item codes should also be 4-letter codes derived from item names in column C. They should also be formatted like category codes but in lower case. For example, Mouthguard should have the code *mout*, and Abdominal Guard should be encoded as *abdg*. It is acceptable if formulas written return 3-letter item codes for items with 3-letter words.
- iii. **Column J (Stock code):** Stock codes should be a concatenation of IDs (column A), category codes (column H), and item codes (column I), with underscores between them. For example, stock code for the first item should be *(5107)_GOLF_bag*.
- iv. **Column K (Stay):** The Stay for each item should show the number of days that items have been in stock. They should be derived by calculating the number of days from the date of delivery in columns D, E, F to the present day. Use the Excel functions Days(), TODAY(), and DATE() to do the calculations.
- v. **Column L (Discount):** Percentage discounts should be applied to various items according to their Stay in column K. Items that have been in stock for up to 30 days should have no (0%) discount. Items that have been in stock for over 30 and up to 60 days should have a 10% discount, while those that have stayed over 60 days should be given discounts of 20%. Use logic IF() function.
- vi. **Column M (Adjusted price):** Calculate the price of all items after the application of the discount in column L. It should be $Price - Discount * Price$.
- vii. **Column N (Tax):** Sales tax on all items should be 13% of adjusted prices in column M.
- viii. **Column O (Sale price):** Add an item's adjusted price to the tax to obtain its sale price.

Note: In case you get unexpected values or #SPILL! error in the respective cells, add the implicit intersection operator (@) in front of range names. Example: item with the implicit intersection operator is @item. You can learn more about this operator [here](#)].

The completed worksheet should resemble Figure 2. A few headings shown here may differ slightly from what you'll create).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	ID	Sport	Item	Day	Month	Year	Price	Category code	Item code	Stock Code	Stay	Discount	Adjusted price	Tax	Sale Price
2	5107	Golf	Bag	28	8	2021	\$199.99	GOLF	bag	(5107)_GOLF_bag	49	10%	\$179.99	\$23.40	\$203.39
3	4460	Golf	Cart	12	12	2020	\$1,899.00	GOLF	cart	(4460)_GOLF_cart	308	20%	\$1,519.20	\$197.50	\$1,716.70
4	5846	Golf	Shoe	12	2	2021	\$89.99	GOLF	shoe	(5846)_GOLF_shoe	246	20%	\$71.99	\$9.36	\$81.35
5	9922	Golf	Balls	21	9	2021	\$29.99	GOLF	ball	(9922)_GOLF_ball	25	0%	\$29.99	\$3.90	\$33.89
6	7793	Golf	Glove	12	2	2021	\$47.85	GOLF	glov	(7793)_GOLF_glov	246	20%	\$38.28	\$4.98	\$43.26
7	9819	Golf	GPS Watch	6	2	2021	\$350.97	GOLF	gpsw	(9819)_GOLF_gpsw	252	20%	\$280.78	\$36.50	\$317.28
8	3111	Golf	Bottom	30	9	2021	\$59.99	GOLF	bott	(3111)_GOLF_bott	16	0%	\$59.99	\$7.80	\$67.79
9	5753	Golf	Head covers	8	8	2021	\$28.89	GOLF	heac	(5753)_GOLF_heac	69	20%	\$23.11	\$3.00	\$26.12
10	1462	Golf	Outerwear	20	9	2021	\$100.00	GOLF	oute	(1462)_GOLF_oute	26	0%	\$100.00	\$13.00	\$113.00
11	7805	Golf	Top	23	8	2021	\$28.99	GOLF	top	(7805)_GOLF_top	54	10%	\$26.09	\$3.39	\$29.48
12	6411	Tenni	Racquet	30	1	2021	\$89.99	TENN	racq	(6411)_TENN_racq	259	20%	\$71.99	\$9.36	\$81.35
13	8705	Tennis	Shoe	10	9	2021	\$99.99	TENN	shoe	(8705)_TENN_shoe	36	10%	\$89.99	\$11.70	\$101.69
14	9882	Tennis	Bag	21	11	2020	\$79.99	TENN	bag	(9882)_TENN_bag	329	20%	\$63.99	\$8.32	\$72.31
15	9353	Tennis	String	15	8	2021	\$19.99	TENN	stri	(9353)_TENN_stri	62	20%	\$15.99	\$2.08	\$18.07
16	4679	Tennis	Balls	14	10	2021	\$17.99	TENN	ball	(4679)_TENN_ball	2	0%	\$17.99	\$2.34	\$20.33
17	1003	Tennis	Top	12	2	2021	\$49.99	TENN	top	(1003)_TENN_top	246	20%	\$39.99	\$5.20	\$45.19
18	5859	Tennis	Polo	2	10	2021	\$89.97	TENN	polo	(5859)_TENN_polo	14	0%	\$89.97	\$11.70	\$101.67
19	2969	Tennis	Short	16	10	2021	\$55.99	TENN	shor	(2969)_TENN_shor	0	0%	\$55.99	\$7.28	\$63.27

Figure 2: Completed Inventory worksheet

Requirements for Part 1 – What we will look for during grading

- a) Worksheet correctly named
- b) Range names used in all formulas
- c) Formatting of columns G, L, M, N, and O
- d) Formulas being general, working automatically for new items, and producing desired results

Part 2: Recurrence Formulas

In this part of the assignment you will be developing a PID control loop algorithm for controlling the state of various systems, from industrial processes to some common household appliances.

PID control¹

For this assignment, we will use a simplified example to illustrate the need for such an algorithm. Consider a common appliance, such as an espresso maker². The espresso machine passes a small amount of hot water through finely ground coffee beans at a controlled rate, to obtain a concentrated beverage that can either be consumed as is, or used for various drinks, like cappuccino. Many parameters influence the quality of the result: bean type and freshness, roast level, water temperature, water flow rate, the amount of water, the type of water, i.e., the dissolved minerals, and so on. Here, let's focus just on the temperature.

Motivation

The optimal water temperature for making espresso is about 88 °C. Which means, one cannot just let the water reach boiling, as it would become too hot (the drink will be too bitter, and some of the flavourful compounds will be destroyed). The most common method for controlling the temperature in inexpensive machines is a simple thermal switch: temperature low – boiler on, temperature high – boiler off. The advantage of this method is its simplicity, and the implementation is rather trivial. However, this results in a very unstable temperature, partially due to the fact that the temperature sensed by the switch lags behind the true temperature of the water and because of the inertia of the system overall. In the Figure 3 below, the desired temperature is 90 °C, and the boiler power switches between either 300 or 600 watts and zero (on-off)³.

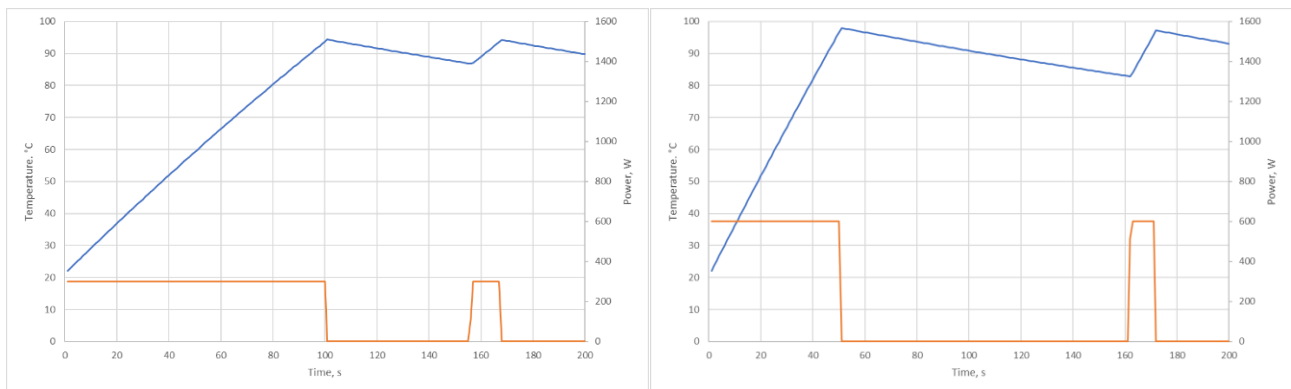


Figure 3: Temperature control using a thermal switch (left: 300 W power; right: 600 W)

One can see that the temperature fluctuates between about 85 degrees and 95 degrees for the lower power rating, and it takes almost two minutes to preheat the water initially. Increasing the power to shorten the preheat time causes even more severe temperature swings. Thus, the quality of the beverage would suffer. At this point, one can start thinking, what if we keep the power high at the beginning, and then gradually reduce it as we approach the desired temperature? That's exactly the idea behind the PID algorithm. The resulting graph would then look similar to the picture below:

¹ For those interested in details, feel free to explore https://en.wikipedia.org/wiki/PID_controller. Here, we will use a rather simplified description of the idea.

² <https://en.wikipedia.org/wiki/Espresso>

³ The model also accounts for heat losses, as can be seen as the negative slope of the blue temperature line.

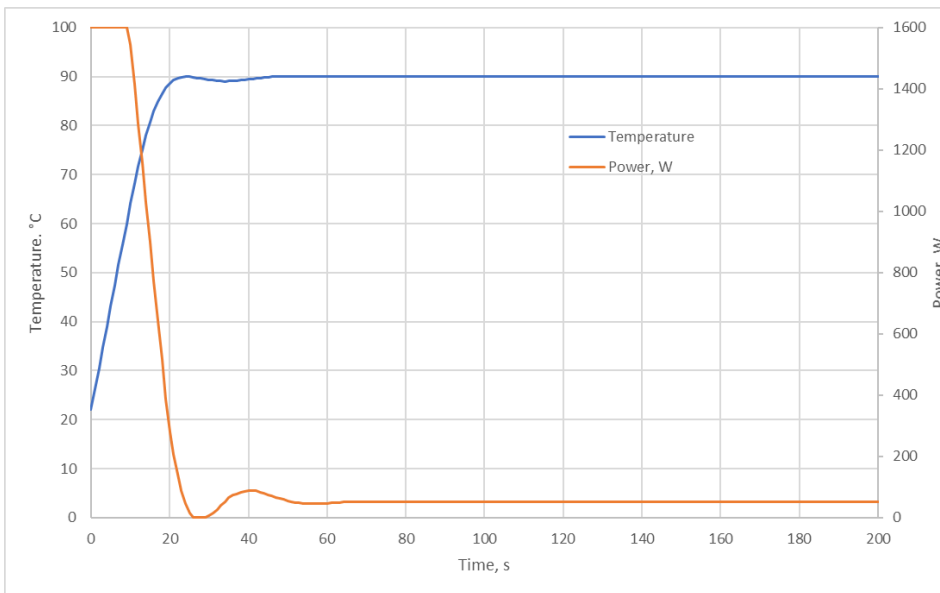


Figure 4: PID control result (1600 W max power)

First, the desired temperature can be reached much faster, as the higher power settings now do not cause any significant overshoots. Furthermore, the temperature stays very stable. However, to achieve this all, one must be able to vary the boiler power smoothly (similar to how one can control the lights with a dimmer switch), and do so based on an output of a computer algorithm. Recent advances in electronics allow for both via approaches like PWM power control and use of programmable microcontrollers. The algorithm does depend on several case-specific parameters, and in this assignment, you will simulate the algorithm and determine those parameters via trial and error. Doing this analytically involves dealing with differential equations, and is not recommended to anyone who values their time.

PID Algorithm

As the name suggests, the algorithm combines three different methods of control. These are *proportional*, *integral* and *derivative*.

Proportional: the amount of correction is proportional to the difference between the desired setpoint (88 °C in our case) and the current state. The colder the water, the more heat would be applied.

Integral: increases the action in relation not only to the error but also the time for which it has persisted. Thus, if the temperature were *slightly* too low, it might not get corrected, unless such an error persists for some time. This integral term is important to reduce the systematic errors, which the other two methods would leave at the end.

Derivative: is mostly responsible for reducing the overshoots. For example, if you approach the set point very quickly, the amount of correction is reduced.

Once the three outputs are added, each with some coefficients, the overall result is going to determine how to bring the system state closer to the setpoint. In our case: what amount of power should be applied at any given moment in order to bring the water temperature to 88 °C as quickly as possible and keep it there as consistently as possible.

Excel Implementation

Step 1: preparing the worksheet

- In the file from Part 1, create a new worksheet and name it *PID control*
- Write these headings on row 1 of the worksheet:

Time, s

Treal, °C

Tsensor, °C

Power, W

P

I

D

	A	B	C	D	E	F	G
1	Time, s	Treal, °C	Tsensor, °C	Power	P	I	D

- Create a Time series in the first (Time) column with an initial value of 0, step size of 1, and stop value of 200.
- Copy the contents of Table 1 below onto cell range J2 to L13 of your worksheet and name the parameters **Pmax**, **TargetT**, **IntitialT**, **RoomT**, **HLRate**, **SHC**, **mass**, **Pfactor**, **Ifactor**, **Dfactor**, **delay**. The grayed-out values are the ones you will need to determine in this part of the assignment.

Table 1: Worksheet parameters

Parameters		
Pmax	1600 W	
Target temp	90 °C	
Initial temp	22 °C	
Room temp	22 °C	
Heat loss rate	0.002 K/K/s	
Specific HC	4184 J/kg/K	
Mass	0.09 kg	
Pfactor		
Ifactor		
Dfactor		
Delay	10 s	

Step 2: Formulas

- Make cell B2 equal to the initial temperature: **=IntitialT**
- Write formulas in cells B3, C2:G2 to estimate the temperature, power setting and the *P*, *I*, and *D* components for the whole time under consideration
- Temperature in column B. The real temperature depends on the heating power that was applied to the amount of water (0.09 kg in the table above) in the last second, and on the cooling rate (in this example, for a room temperature of 22 degrees and the starting temperature of 72 degrees, the temperature will drop by $0.002 \times [72 - 22] = 0.1$ °C).
In case you need to refresh your physics, one watt is defined as one joule of energy per second, and the temperature increase can be derived from the following relationship:

$$Q = mc\Delta T$$

Q = heat energy

m = mass

c = specific heat capacity

ΔT = change in temperature

Overall, the result will depend on the temperature and the power value in the line *directly above* the current line.

Hint: use explicit cell references to construct your formula

- Temperature in column C. The temperature measured by a temperature sensor, later used as an input for the PID algorithm, depends on the inertia in the system. Here, it's simply the values from column B **delayed** by some number of seconds (or rows). There are multiple ways to implement it; however, the behaviour of the values should depend on the Delay value in the parameters

Hint: consider using functions like INDEX, MAX, ROW, etc.

B	C
Treal, °C	Tsensor, °C
22	22
26.2489909	22
	22
	22
	22
	22
	22
	22
	22
	22
	22
	22
	26.2489909

Figure 5: The temperature columns. Your actual values may be different

- **P** column. P column is simply the difference between the target temperature and the measured temperature.
- **I** column. I column is similar to the previous one, except the values are accumulated. The formula in row 2 will probably be different from the one in the subsequent rows. As earlier, explicit cell references may be needed to construct your formula.
- **D** column. D column is rate of temperature change, with a negative sign. Explicit cell references may be needed here as well.
- For the column D (with the heading **Power, W**), compute the result by weighting the **P**, **I**, and **D** columns with their respective coefficients. Consider using functions such as MIN and MAX to constrain the power values to a range from zero to **Pmax**.
- Remember to use named cells and ranges whenever possible, that is, whenever the use of explicit references is not required to solve a task.
- Fill in some initial parameters for the three components. Consider starting with only **Pfactor** being non-zero.

At the end, your worksheet will contain values resembling the following:

	A	B	C	D	E	F	G	H	I	J	K	L
1	Time, s	Treal, °C	Tsensor, °C	Power, W	P	I	D					
2	0	22	22	1600	66	66	0			Parameters		
3	1	26.2489909	22	1600	66	132	0			Pmax	1600 W	
4	2	30.4894837	22	1600	66	198	0			Target temp	88 °C	
5	3	34.7214956	22	1600	66	264	0			Initial temp	22 °C	
6	4	38.9450435	22	1600	66	330	0			Room temp	22 °C	
7	5	43.1601443	22	1600	66	396	0			Heat loss rate	0.002 K/K/s	
8	6	47.3668149	22	1600	66	462	0			Specific HC	4184 J/kg/K	
9	7	51.5650721	22	1600	66	528	0			Mass	0.09 kg	
10	8	55.7549328	22	1600	66	594	0			Pfactor		
11	9	59.9364138	22	1600	66	660	0			Ifactor		
12	10	64.1095319	22	1600	66	726	0			Dfactor		
13	11	68.2743037	26.2489909	1396.507	61.75101	787.751	-4.24899			Delay	10 s	
14	12	71.890346	30.4894837	1261.986	57.51052	845.2615	-4.24049					

Figure 6: Worksheet after all the formulas entered

Step 3: Visualizing the data

- Plot the temperature and the power values in a graph, similar to Figure 4. You must use different line colours than those shown in Figure 4. Insert the legend and label the axes appropriately.
- Position your graph directly below the parameters and make it about 8–10 columns wide.
- After this, adjust the values to reach the look resembling Figure 4 as closely as possible. You will notice that very small changes of factors can have dramatic results; be patient. Your submission should contain the values that provide the best temperature control.

Requirements for Part 2 – What we will look for during grading

- Worksheet correctly named
- PID data and parameters table formed as shown. The PID data and graph should respond to changing values in the parameters table. Your own choice of parameter values
- Range names assigned and used in formulas whenever possible
- Correct formulas producing correct results
- Graph with appropriate labels, legends, differently coloured lines, and positioned just underneath the parameters table