PACCMAN Developer's Guide:

(Python Analysis of Conformal Cooling Molds And/or desigNs)

This guide is meant as a resource for using, adding to, or integrating the PACCMAN program. Feedback is welcome. Please report all bugs to hughfeehan353 on Github. Thanks.

Introduction:

In its current state, the program has three options presented to the user.

Selecting "N" uses the first material properties (KM1, etc.) and then gives choice of heat transfer coefficient correlation.

Selecting "M" compares the three materials using the Gnielinski correlation.

Selecting "H" compares the three heat transfer coefficient correlations using the first material property (KM1, etc.).

For all three choices, the program asks the user if they would like to save graphs of the average heat cycle temperature over time.

Choosing "Y", downloads the graph in .png and .eps formats.

The program is written so that by changing the variables, it can analyize any conformal cooling design.

The printed out data is as follows:

flow velocity, kinematic viscosity, Reynolds number, Prandl number, Farcy friction factor, heat transfer coefficient, average heat cycle temperature of the mold, time constant, and coolant pressure drop.

The program can by testing by running pytest in its directory.

Application Fundamentals:

At the beginning, the program initializes and assigns all the basic variables. These can all be changed, or the list can be added to in order to improve the function of the program. However, they are all necessary for the program to function properly so should not be completely omitted.

Afterwards, all the necessary functions are defined. Similarly, these cannot be removed as they are necessary for the program’s function but could be added to.

At this point there is an “if” statement to check for the user’s desired function for the program. This “if” statement contains all of the program’s calculation functions in order to ensure that nothing will break if the user chooses an invalid letter.

The first thing inside the “if” statement are some calculations that must be completed no matter the chosen choice of the program’s function.

The rest of the calculations have nested “if” and “elif” statements so that the correct variables are used for the chosen program function.

Once the calculations are completed, the program creates a plot of the average heat cycle temperature over time that the user has the choice of downloading.