|  |  |  |
| --- | --- | --- |
| INPUT | PROCESSES | OUTPUT |
| 1. Option from main menu | * Convert the input to integer * If the input is not a number, print out the error and ask for input again * Check which action to perform with the chosen option | * Print out the menu for the chosen action * Print out error if encountered any * Quit if chosen quit |
| 1. Dimensions required in the diamond drawing menu | * Perform operations depending on the dimensions to determine what to print | * Diamonds with the given dimensions * Main menu after finished drawing |
| 1. Number representing the chosen category in the CTF menu | * Convert the input to integer * If the input is not a number, print out the error and ask for input again * Check which action to perform with the chosen option | * Challenges within the chosen category * Error if encountered any * Main menu if chosen to return to main menu |
| 1. Flag submission | * Check whether the flag for that particular challenge is correct. If it is, check if challenge is already solved. * If the challenge is already solved, print out some messages. * If not, add the challenge to the solved\_challenges list then add some points to the user * If the flag is incorrect, print out some messages. | * Messages about the flag submission |
| 1. User’s name | * Store it in a variable to print out | * Main menu after inputting name |

I’ve chosen the variable **points** to be **global and integer.** I did this so that I could manipulate the variable’s value throughout many functions, and integer to perform arithmetic operations on the variable.

I’ve chosen the variable **option** to be an **integer** not string because there could be 2 possible errors: invalid type and invalid value. Invalid type happens when the user enters a character rather than a number. Invalid value happens when the user enters a value that is either too great or too small. Therefore, using integer, combining with the error handler try, except I could determine the error and handle it correctly.

I use functions to cut down my code to smaller chunks. Easier to read and debug.

I’ve chosen the variable **answer** to be a string because the flag is a combination of letters, numbers and signs.

The dimensions of the diamonds are integers so that I could perform arithmetic operations on them.