Jonathan Antenanie
COMP2401 A3

Design Analysis Document for Assignment three

- We are tasked to do the following things.
  1. Load CSV

     readCSV takes a constant file name.
     a. error handling message
        i. if not found error
        ii.
     b. Then we create a Data type

        We will use this to load the csv.
     c. By using getline(),
        i. we get every line of file.
        ii. we add it to PokemonData we created
     d. Read the entire file into Pokemon Data
        i. Bytes read.
        ii. Update total size.
        iii. Close

  2. Creating pokemon

     Takes a csv data and a type of match to search.
     a. Initialize the pokemon array.
     b. Using strok and a helper function phrase.
        i. Phrase just phrases the char return from getline.
     c. Then we call strcmp to compare them
        i. If they are the same, we add it to the pokemon array we initialized.

  3. Add pokemon to Array.
     a. Takes an Array and a Pokemon struct.
        i. It automatically checks and resizes the array.
           1. Uses helper function resize_array().
        ii. It adds the pokemon to the array->data.

The main function
In the men function, we will do the following and make use of almost all of functions we defined in the library!

1. Initialize the variable.
   a. PokemonArray array.

2. Prompt the user to enter the name of the file containing the Pokemon descriptions.
   - Read the filename from the user input.
   - calls one of the functions we defined to do so.

3. Try to read the CSV file using the readCSV function.
   a. If the file is not found, display an error message.
   b. Ask the user to enter the name again or exit the program.

```
        c. If the file is found, continue to the next step

   4. Initialize a mutex for synchronization.
   5. Display the menu options in a loop.
        a. Print the menu options to the console.
        b. Read the user's choice (a, b, or c) from the input.
   6. Inside the loop, handle the user's choice:
        a. If the user chooses 'a' (Type search):
              i.  Prompt the user to enter the Type1 property to search.
             ii.  Create the necessary arguments to pass and
                  search_pokemon_threaded function.
            iii.  Create a new thread for the search operation using
                  pthread_create.
             iv.  Continue to the next iteration of the loop.

        b. If the user chooses 'b' (Save results):
              i.  Prompt the user to enter the name of the file to save
                  the results.
             ii.  Create the necessary arguments to pass to the
                  save_results_threaded function.
            iii.  Create a new thread for the save operation using
                  pthread_create.
        c. Or the user chooses to exit.
   7. Free every Memory
   8. Destroy mutex
```

Note that, to show I am doing the searching and adding concurrently with the main, I added a debug print function that will show up when displaying the menu. It is just one line code!