PPS

PPS
↓
user enter path → error → reenter / exit
↓
start connection with [PQC]
↓
receive from PQC ← sent ← [PQC]
↓
response → search / save / exit

Server

client

[PPS.C]
↓
[connection] → connect → [response]
↓
[response] ← type save ← Exit

[PQS.c]
↓
[response]

```c
//create two global variable

ArrayList * array_to_be_saved;

FILE * file_to_be_saved;

char file_names[1024] = "0";

int main() {

    //line 14 to 41 is to initialize server end (number may change after some modifications)

    //initialize ip address and port number

    //initialize server and client address

    //using tcp

    //line 44 to 52 below are going to bind server

    //line below is linten

        send(client_sock, &buffer, strlen(buffer), 0);

FILE * step1_read_file(FILE * fptr){

    //read the path/name

    //verify if the file exists or path is correct

        //empty the buffer

        //first verify if the user want to exit or continue

ArrayList * read_pokemons(FILE* fptr, ArrayList * arrayList) {

    }

/*

 * search pokemon by user choice

 */

ArrayList * search_pokemon(ArrayList * arrayList, char* type){

void save_pokemon(char * file_name){

    }

}
```

I used 3 header file, array_list.h defines the pointer, size, max size and the size for each item. It also claims the signature for four methods, alist_initialize, alist_add, alist_clear and alist_get.

PPS.h defines the struct PokemonType and four signatures for methods, including read file, read pokemon, search pokemon and save pokemon. PQC.h includes all needed header for the client.

In terms of running this program, please open two cmd, one for the server and another for the client.