```
qin@linprog8.cs.fsu.edu:~>clear
```

```
qin@linprog8.cs.fsu.edu:~>~vastola/usub/submit1 poke.cpp
********************* File submitted **********************
   Here are the contents of your submitted file:
***********************************************************
/* Name: Caijun Qin
Date: 12/2/2018
Section: 6
Assignment: 7
Due Date: 12/7/2018
About this project: This is a simplified down game of Pokemon Go. The player has
30 chances to catch a pokemon from any given pokedex list. The player has access
to three kinds of balls that each give a different bonus boost to the chance of
catching a pokemon. Each ball can be utilized 10 times.
Assumptions: The textfile name and the trainer's name will not exceed 40 spaces.
All work below was performed by Caijun Qin */

#include <iostream>
#include <iomanip>
#include <string>
#include <cstring>
#include <cctype>
#include <cstdlib>
#include <ctime>
#include <fstream>
using namespace std;

//STRUCTURES
typedef struct Pokemon {
    int index;
    string name;
    string type;
    int catchPercent;
    int numCaught;
    int numSeen;

} POKEDEX;

//FUNCTION DECLARATIONS
void printMenu();

int main(){
    //prompts user for name of text file to open
    string textFile = "poke.txt";
    cout << "Enter the name of the file to open > ";
    getline(cin, textFile);

    //attemps to open the file
    ifstream input;
    input.open(textFile.c_str());

    //checks if file actually exists and in the same directory as "poke.cpp"
    bool fail = input.fail();
    while(fail != false){
        cerr << "Invalid or nonexistent file. Please try again. \t";
        getline(cin, textFile);
        input.open(textFile.c_str());
        fail = input.fail();
    }

    //makes the dynamic array for "Pokemon" struct
    int inputSize = 0;
    input >> inputSize;
    POKEDEX * list = new POKEDEX[inputSize];

    //appends pokemon info from the input file "poke.txt" to the array "list"
    for(int counter = 0; counter < inputSize; counter++){
        input >> list[counter].index;
        input >> list[counter].name;
        input >> list[counter].type;
        input >> list[counter].catchPercent;
```

```
    list[counter].numCaught = 0;
    list[counter].numSeen = 0;

}

input.close();
cout << "\n\n";

//game officially starts
//general variables
string trainer = "";
char menuChoice = ' ';

//ball types and counts
int pokeBall = 10;
int greatBall = 10;
int ultraBall = 10;
int ballChoice = 0;

//trainer and game instructions
cout << "What's your name, trainer? > ";
getline(cin, trainer);
cout << "\n";

cout << "Welcome, " << trainer << ", to the COP3014 Safari Zone! \n";
cout << "You'll have 30 chances to catch Pokemon, make them count! \n";
cout << "The game will end once you run out of balls or if you choose to quit early \n\n";

while(menuChoice != 'Q'){
    printMenu();
    cin.get(menuChoice);
    cout << "\n\n";

    //executes function based on menuChoice
    switch(menuChoice){
        //plays the game
        case '1' :
            {
            //finds a random pokemon from POKEDEX
            srand((unsigned int) time(0));
            int index = (rand() % inputSize) + 1;
            string seenPokemon = list[index - 1].name;
            list[index - 1].numSeen++;

            //options and outcomes for a specific pokemon
            cout << "A wild " << seenPokemon << "has appeared! \n\n";
            cout << "Choose a ball to throw: \n";
            cout << "1 - Poke Ball  (You have: " << pokeBall << ") \n";
            cout << "2 - Great Ball (You have: " << greatBall << ") \n";
            cout << "3 - Ultra Ball (You have: " << ultraBall << ") \n";
            cout << "Selection > ";
            cin >> ballChoice;
            while(ballChoice != 1 && ballChoice != 2 && ballChoice != 3){
                cerr << "Invalid Selection. Try again. \n";
                cin >> ballChoice;
            }
            switch(ballChoice){
                case 1 :
                    if(pokeBall > 0){
                        cout << "Threw a Poke Ball! \n";
                        srand((unsigned int) time(0));
                        int randNum = (rand() % 100) + 1;
                        int catchRate = list[index - 1].catchPercent;
                        int totalCatchRate = 0;
                        pokeBall--;
                        totalCatchRate = catchRate;
                        if(randNum <= totalCatchRate){
                            list[index - 1].numCaught++;
                            cout << "Congratulations! You caught " << seenPokemon << "! \n\n";
                        } else {
                            cout << seenPokemon  << " ran away. :-( \n\n";
                        }
                    } else {
                        cerr << "You have none of them! \n\n\n";
                    }
                    break;
                case 2 :
                    if(greatBall > 0){
                        cout << "Threw a Great Ball! \n";
                        srand((unsigned int) time(0));
                        int randNum = (rand() % 100) + 1;
                        int catchRate = list[index - 1].catchPercent;
                        int totalCatchRate = 0;
                        greatBall--;
                        totalCatchRate = catchRate + 20;
                        if(randNum <= totalCatchRate){
                            list[index - 1].numCaught++;
                            cout << "Congratulations! You caught " << seenPokemon  << "! \n\n";
                        } else {
                            cout << seenPokemon  << " ran away. :-( \n\n";
                        }
                    } else {
                        cerr << "You have none of them! \n\n\n";
                    }
                    break;
                case 3 :
                    if(ultraBall > 0){
                        cout << "Threw an Ultra Ball! \n";
                        srand((unsigned int) time(0));
                        int randNum = (rand() % 100) + 1;
                        int catchRate = list[index - 1].catchPercent;
                        int totalCatchRate = 0;
                        ultraBall--;
                        totalCatchRate = catchRate + 40;
                        if(randNum <= totalCatchRate){
                            list[index - 1].numCaught++;
                            cout << "Congratulations! You caught " << seenPokemon  << "! \n\n";
                        } else {
                            cout << seenPokemon  << " ran away. :-( \n\n";
                        }
                    } else {
                        cerr << "You have none of them! \n\n\n";
                    }
```

```
    }
    cin.ignore();
    }
    break;
case '2' :
    {
    //accumulation variables for catch statistics
    int totalCaught = 0;
    int totalSeen = 0;
    cout << "POKEDEX: \n";

    //printing out the POKEDEX stats
    for(int counter = 0; counter < inputSize; counter++){
        totalCaught = totalCaught + list[counter].numCaught;
        totalSeen = totalSeen + list[counter].numSeen;
        cout << "no " << left << setw(10) << counter + 1;
        cout << left << setw(20) << list[counter].name;
        cout << left << setw(10) << "Caught:  " << list[counter].numCaught << "  ";
        cout << left << setw(10) << "Seen:  " << list[counter].numSeen << "\n";
    }
    cout << "Total Pokemon Caught: " << left << setw(10) << totalCaught;
    cout << "Total Pokemon Seen: " << totalSeen << "\n";
    cout << fixed << showpoint << setprecision(2) << "Overall Catch Rate: " << ((double) totalCaught) / totalSeen * 100 << "%";
    cout << "\n\n";
    }
    break;
case '3' :
    cout << "You have:  \n";
    cout << pokeBall << " PokeBalls \n";
    cout << greatBall << " Great Balls \n";
    cout << ultraBall << " Ultra Balls \n";
    cout << "\n\n";
    break;
case '4' :
    {

    cout << "\n";
    //variables
    int arrIndex = 0;

    //GRASS
    cout << "Grass:  ";
    for(int counter = 0; counter < inputSize; counter++){
        string a = list[counter].type;
        string b = list[counter].name;
        if(a == "Grass" || a == "grass" || a == "GRASS"){
            if(arrIndex == 0){
                cout << b;
                arrIndex++;
            } else if(arrIndex > 0){
                cout << ",  " << b;
                arrIndex++;
            }
        }
        if(counter == inputSize - 1){
            arrIndex = 0;
            cout << "\n\n";
        }
    }

    //FIRE
    cout << "Fire:  ";
    for(int counter = 0; counter < inputSize; counter++){
        string a = list[counter].type;
        string b = list[counter].name;
        if(a == "Fire" || a == "fire" || a == "FIRE"){
            if(arrIndex == 0){
                cout << b;
                arrIndex++;
            } else if(arrIndex > 0){
                cout << ",  " << b;
                arrIndex++;
            }
        }
        if(counter == inputSize - 1){
            arrIndex = 0;
            cout << "\n\n";
        }
    }

    //WATER
    cout << "Water:  ";
    for(int counter = 0; counter < inputSize; counter++){
        string a = list[counter].type;
        string b = list[counter].name;
        if(a == "Water" || a == "water" || a == "WATER"){
            if(arrIndex == 0){
                cout << b;
                arrIndex++;
            } else if(arrIndex > 0){
                cout << ",  " << b;
                arrIndex++;
            }
        }
        if(counter == inputSize - 1){
            arrIndex = 0;
            cout << "\n\n";
        }
    }

    //BUG
    cout << "Bug:  ";
    for(int counter = 0; counter < inputSize; counter++){
        string a = list[counter].type;
        string b = list[counter].name;
        if(a == "Bug" || a == "bug" || a == "BUG"){
            if(arrIndex == 0){
                cout << b;
                arrIndex++;
            } else if(arrIndex > 0){
                cout << ",  " << b;
                arrIndex++;
            }
```

```
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//NORMAL
cout << "Normal:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Normal" || a == "normal" || a == "NORMAL"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//POISON
cout << "Poison:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Poison" || a == "poison" || a == "POISON"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//ELECTRIC
cout << "Electric:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Electric" || a == "electric" || a == "ELECTRIC"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//GROUND
cout << "Ground:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Ground" || a == "ground" || a == "GROUND"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//FAIRY
cout << "Fairy:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Fairy" || a == "fairy" || a == "FAIRY"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//FIGHTING
cout << "Fighting:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Fighting" || a == "fighting" || a == "FIGHTING"){
```

```
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//PSYCHIC
cout << "Psychic:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Psychic" || a == "psychic" || a == "PSYCHIC"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//ROCK
cout << "Rock:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Rock" || a == "rock" || a == "ROCK"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//GHOST
cout << "Ghost:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Ghost" || a == "ghost" || a == "GHOST"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//ICE
cout << "Ice:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Ice" || a == "ice" || a == "ICE"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}

//DRAGON
cout << "Dragon:  ";
for(int counter = 0; counter < inputSize; counter++){
    string a = list[counter].type;
    string b = list[counter].name;
    if(a == "Dragon" || a == "dragon" || a == "DRAGON"){
        if(arrIndex == 0){
            cout << b;
            arrIndex++;
        } else if(arrIndex > 0){
            cout << ",  " << b;
            arrIndex++;
        }
    }
    if(counter == inputSize - 1){
        arrIndex = 0;
        cout << "\n\n";
    }
}
```

```
                }
            }
    }

    return 0;
}


//FUNCTION DEFINITIONS
void printMenu(){
    cout << "1 - Hunt for a Pokemon! \n";
    cout << "2 - Pokedex Statistics. \n";
    cout << "3 - Print Inventory. \n";
    cout << "4 - List Pokemon By Type \n";
    cout << "Q - Quit \n";
    cout << "Selection > ";

    return;
}




********************* END FEEDBACK **********************
*    PLEASE VIEW ABOVE TO VERIFY THE CONTENTS OF         *
*              YOUR SUBMITTED FILE                       *
***********************************************************
qin@linprog8.cs.fsu.edu:~>
```