

CSCI 1300 CS1: Starting Computing

Instructor: Ashraf/Corell/Cox/Fleming, Fall 2019

Project 3: Choose Project, Meet with TA/CA to go over design

Before Wednesday November 13

Project 3: Submit Class files & Code Skeleton

Due Wednesday November 13 by 11 PM (no early submission bonus)

Project 3: Final Deliverables - Due Wednesday December 4 by 11 PM (no early submission bonus)

Project 3: Project Report - Due Sunday December 8 by 11 PM (no early submission bonus)

Project 3: Interview Grading

Begins on Thursday, December 5. Must be completed at the latest on Friday December 13.

This project is worth 15% of your overall course grade.

The minimum requirements for the project can be found at the end of the project description.

“Charmanders Are Red. Squirtles Are Blue.

If You Were A Pokemon, I'd Choose You.”

For the Final Project we will implement an interactive board game fashioned after the game **Pokémon**. The user plays the role of a Trainer in a quest through the vast Pokémon world to find and train magical creatures called Pokémon. As the player acquires more Pokémon, they also have to defeat the other Trainers, randomly scattered throughout the land. The goal of the game is to find, battle, and defeat the other Trainers by growing and evolving their suite of Pokemon

The world is represented by a rectangular board consisting of 25 rows by 16 columns of tiles, where each tile is a location in the vast Pokémon terrain. The map can be found in the file `mapPoke.txt`.

The board is comprised of 400 tiles as follows:

- 153 water tiles (marked with a 'w'),
- 247 land tiles, marked as follows:
 - Plain land tiles are marked with a 'p'
 - 15 Pokémon Centers are marked with a 'C'
 - 15 Pokémon Gyms are marked with a 'G'



[illegible]

The file `pokemon.txt` has name, stats and type about each of the possible 151 Pokémon. Below, we have included the first 11 lines in the file. Notice that some Pokémon have more than one type.

#	Pokémon	HP	Attack	Defense	Speed	Max	Type	
1	Bulbasaur	45	49	49	45	65	Grass	Poison
2	Ivysaur	60	62	63	60	80	Grass	Poison
3	Venusaur	80	82	83	80	100	Grass	Poison
4	Charmander	39	52	43	65	50	Fire	
5	Charmeleon	58	64	58	80	65	Fire	
6	Charizard	78	84	78	100	85	Fire	Flying
7	Squirtle	44	48	65	43	50	Water	
8	Wartortle	59	63	80	58	65	Water	
9	Blastoise	79	83	100	78	85	Water	
10	Caterpie	45	30	35	45	20	Bug	

All Pokémon have five statistics which affect their performance in battle. These are **HP**, **Attack**, **Defense**, **Max Attack/Defense**, and **Speed**. Some of these statistics can be modified during the course of the game.

- **HP** (short for Hit Points): A Pokémon faints when its HP reaches zero, and it cannot be used in battle again until it is revived at a Pokémon Center.
- **Attack**: Determines the strength of a Pokémon's attacks.
- **Defense**: Determines the Pokémon's resistance against attacks.
- **Max Attack/Defense**: Determines the maximum Pokémon's strength/resilience value as they level up.
- **Speed**: After battle commands are entered, the Speed statistics of the participating Pokémon are compared. The Pokémon with higher Speed will attack before those with lower Speed.

The player begins **The Pokémon Game** with one starter Pokémon, which they have received from that region's Pokémon Professor. The starter Pokémon will become the *Active* Pokémon in the player's party. The professor offers four Pokémon choices, and the player must choose one from the following: *Bulbasaur*, *Charmander*, *Squirtle* and *Pikachu*. The starting location of the player Trainer is conditioned by their Pokémon choice:

1. If they choose *Bulbasaur*, they start at row 13, column 7
2. If they choose *Charmander*, they start at row 13, column 8
3. If they choose *Squirtle*, they start at row 13, column 9
4. If they choose *Pikachu*, they start at row 13, column 10

At the beginning of the game, the player starts with **10 Pokéballs**. Pokéballs function as currency during the game; the player spends it to heal the Pokémon in their party, or during battles, and they gain it back if they are successful in their encounters.

There are **15** other Trainers. Each Trainer will start the game at one of the specified Pokémon Gym locations on the map. Each Trainer starts with between 1 and 6 different Pokémon, chosen at random at the start of the game. If the player defeats another Trainer, they will acquire that trainer's suite of Pokémon, earn a **badge** from the Gym, **5 Pokéballs**, and **60 points**. The player cannot battle the same Trainer twice. If a player collects 6 badges, they win the game.

Because each player begins the game with only one starter Pokémon, **catching** is one of the most fundamental aspects of the Pokémon game, and the primary method of recruiting new Pokémon to the player's party. At the beginning of the game, twenty (**20**) Pokémon are scattered randomly throughout the land tiles. As the player encounters a wild Pokémon, they can attempt to catch it by battling it (more details about battles with wild Pokémon later in the write-up). For each wild Pokémon caught, the player gains **10 points**, and their supply of **Pokéballs goes down by 1**.

The player has 6 available slots in their Pokémon suite, and only Pokémon from this party can be designated *Active* and used in battles. If the player acquires more than 6 Pokémon, the extra Pokémon will be stored in the Pokédex, which can be accessed at the Pokémon Centers. At any time during the game, if the player has acquired Pokémon of 8 different types, they win the game. If the player quits before acquiring 6 badges or 8 different Pokémon types, they will be ranked given the amount of points acquired up to that point.

“Strong Pokemon. Weak Pokemon. That is only the selfish perception of people. Truly skilled trainers should try to win with all their favorites.”

Start the game by asking the player to choose a starting Pokémon. Initialize all the information associated with the player, including their location.

Here is an example of a potential sample run:

Welcome to Pokémon!

Please state your name: **No 1 Fan**

Welcome, **No 1 Fan**! Before you can begin your Pokémon adventure, you must choose a starting Pokémon, courtesy of the Professor. Please choose from the following Pokémon:

1. Bulbasaur
2. Charmander
3. Squirtle
4. Pikachu



Then, load the given text files and set up the rest of the game, initializing all the Gyms, the Trainers (with their 1-6 random Pokémon), and the Pokémon Centers. Then, scatter the 20 random wild Pokémon across the land.

Note: a wild Pokémon cannot start:

- at the same location as a Pokémon Center
- at the same location as a Pokémon Gym
- at the same location as the main player
- on a tile that is on water
- on a tile that is already occupied by another wild Pokémon

Now is the time to show the map to the player. This occurs at the beginning of each turn. The player has visibility in a 7 x 7 area centered at their current location. For example, if the player has chosen “Pikachu” as their starting Pokémon, the board displayed at the beginning of the game would look like this:

*	C	*	*	*	C	*
*	~	G	*	*	*	*
*	~	~	~	*	G	~
*	*	*	@	*	*	~
*	*	*	C	*	~	~
~	*	*	*	*	~	~
~	*	*	*	*	~	~

Where ‘G’ is a Gym, ‘C’ is a Pokémon Center, ‘@’ is the starting location for the player. All water tiles should be displayed with the symbol ‘~’ and all land tiles with the symbol ‘*’. If the player is near/at one of the edges of the map, you may use other characters to represent the edges/outside of the map region.

“Make your wonderful dream a reality, it will become your truth.

If anyone can, it’s you.”; Turn Options

At the beginning of each turn, after displaying the map, the number of Pokéballs and the stats of all the Pokémon in the player’s party must be printed. Here is an example, right before the beginning of the first turn:



```
* C * * * C *
* ~ G * * * *
* ~ ~ ~ * G ~
* * * @ * * ~
* * * C * ~ ~
~ * * * * ~ ~
~ * * * * ~ ~
```

The sun is shining, **No 1 Fan!** It’s a beautiful day, we have 10 Pokéball left and your Pokémon are strong.

Name: Pikachu (ACTIVE), HP: 35, A: 55, D: 30, S: 90, M: 50

Please choose from the following options:

1. Travel
2. Rest
3. Try your luck
4. Quit the game

After printing the stats, the player is presented with 4 choices:

1. **Travel:** ask the player to pick a travel direction: North, South, East or West.
 - If the desired location is a “water” tile, the player cannot travel there; ask the player to choose again
 - If by choosing a certain direction the new location would be outside the boundaries of the map, ask the player to choose again.
 - If a player chooses to travel, then they may not stay in the same location

If the location the player wishes to travel is a Pokémon Center, follow the procedure for visiting Centers (page. 11)

2. **Rest:** this can be a useful option if the Pokémon in the player’s suite need to heal after a battle.
 - the player’s location remains unchanged
 - each Pokémon in the player’s party increases their HP by 1
 - the number of Pokéballs goes down by 1

3. **Try their Luck:**
 - The player’s location remains unchanged
 - If a wild Pokémon is located within the 7 x 7 area around the player, the player has a 50% chance of catching it for free (the number of Pokéballs remains unchanged)

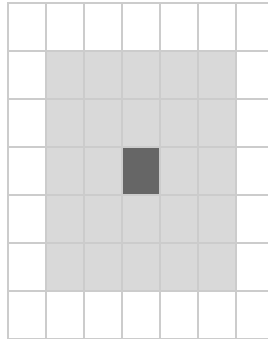
4. **Quit the Game**

After the player’s stats are updated (location, Pokémon HP values, # Pokéballs), the program should update the location of all the remaining wild Pokémon, by randomly moving them one tile, in the North, South, East or West direction. The same rules apply:

- If the new generated random location is a “water” tile, the Pokémon cannot travel there; you will need to generate another random travel direction.
- Pokémon must remain within the boundaries of the map.
- Pokémon cannot travel at the same location as a Pokémon Center
- Pokémon cannot travel at the same location as a Pokémon Gym
- Pokémon cannot travel on a tile that is already occupied by another wild Pokémon
- Pokémon may not stay in the same place; they must move in *some* direction.

“I see now that one’s birth is irrelevant. It’s what you do that determines who you are.”; Wild Pokémon Encounters

After all the locations have been updated, it’s time to resolve possible encounters. Check first if the player is in proximity to other **wild Pokémon**. If a wild Pokémon’s location is within 2 tiles in any direction (including diagonally) of the player’s location (depicted by the shaded region below), then they will have an encounter.



If an encounter with a wild Pokémon occurs, battle information should be displayed, with the player’s *Active* Pokémon, the opposing Pokémon, their respective HP bars, and an option menu (3 choices, see below). At any time, the player may carry up to six Pokémon in his or her party; the *Active* Pokémon is automatically sent into battle. At the start of each battle-turn, the player can choose to **Fight**, or **Switch** the *Active* Pokémon for another in their party, or attempt to **Run** from battle.

You ran into a wild Pokémon!

Name: Weedle, HP: 40, A: 35, D: 30, S: 50, M: 20

Your ACTIVE Pokémon is:

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

What do you want to do (pick 1, 2, or 3):

1. Fight
2. Switch Active Pokémon
3. Run

Each option leads to an action as follows:

1. **Fight:** The player’s *Active* Pokémon will battle the wild Pokémon. Decide first if the wild Pokémon will attack or attempt to flee.



- If this is the first or the second wild Pokémon the player encounters, the wild Pokémon will automatically join the player's active party
- In all other cases, there is a 60% chance the wild Pokémon will attack.

A. If the wild Pokémon decides to flee, proceed as outlined at the **Run** option below (calculate F, ... etc.) to decide if escape by the wild Pokémon is successful. If escape is not successful, then the wild Pokémon will automatically join the player's party. If escape is successful, then the wild Pokémon will teleport to a random location on the board (Note: all previous rules with regards to locations apply).

$$F = \frac{A \times 32}{B} + 30 \times C$$

where:

A is the current Speed of the wild Pokémon, trying to flee

B is the player's *Active* Pokémon's Speed divided by 4, mod 256, and

C is the number of times the wild Pokémon has tried to escape during the encounter (counting the current attempt).

Example: Weedle wants to flee Pikachu

Name: Weedle, HP: 40, A: 35, D: 30, S: 50, M: 20

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

A = 50 (the speed of the Pokémon trying to flee)

B = (90/4)mod 256 = 22.5 (based on the speed of Pikachu, 90)

C = 1 (first attempt to flee)

F = (1600/22.5) + 30 = 101

If F is greater than 255, the wild Pokémon escapes automatically. Otherwise, a random number is generated between 0 and 255. If that number is less than F, the wild Pokémon escapes. If not, the escape fails.

- B. If both sides attack, the one who goes first is determined by *Speed*. The Pokémon with the highest *Speed* value will attack first (i.e. it will be considered **the attacker**) while the other Pokémon will defend (i.e. it will be considered **the defendant**). For the first attack:
1. Generate a random integer value A between 0 and the attacker's *Attack* value.
 2. Generate a random integer value D between 0 and the defendant's *Defense* value.
 3. If A is larger than D, the defendant will suffer (A-D) damage: subtract the damage value from the defendant's HP.

4. If A is smaller than or equal to D, then the defendant incurs no damage.

Example: Weedle and Pikachu both want to attack

Name: Weedle, HP: 40, A: 35, D: 30, S: 50, M: 20

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

Pikachu has a larger value for Speed (90) than Weedle's value for Speed (50). So Pikachu attacks first. Display the first phase of the attack, the damage dealt and the HP left for the defendant.

Pikachu attacks first

Pikachu deals 20 damage

Name: Weedle, HP: **20**, A: 35, D: 30, S: 50, M: 20

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

Do the same procedure for the second attack phase.

Weedle attacks first

Weedle deals 3 damage

Name: Weedle, HP: **20**, A: 35, D: 30, S: 50, M: 20

Name: Pikachu, HP: **32**, A: 55, D: 30, S: 90, M: 50

Outcome 1: If at the end of either attack phase, the defendant Pokémon is damaged to the point where its HP becomes 0, then the defendant Pokémon *faints* and the attacker Pokémon *levels up*.

- A. If the wild Pokémon is the one who fainted,
 - a. its HP is automatically restored
 - b. it will join the player's party
 - c. the *Active* Pokémon who won the battle *levels up*
 - d. **and the turn ends.**
- B. If the player's Pokémon is the one who fainted, then
 - a. The Pokémon who fainted will not be able to be sent into battle, until its HP will be restored during the next visit to a Pokémon Center.
 - b. The wild Pokémon *levels up*
 - c. the dialog goes back to the main battle menu (with the three options: Fight, Switch Pokémon, and Run), which will allow the player the choice to send out another Pokémon to take its place, drawn from his or her party.
- C. If the player's Pokémon is the one who fainted, and if this was the last awake Pokémon in the player's party, then all Pokémon in the party are now fainted.
 - a. The player loses the battle,

- b. The player is teleported to the nearest Pokémon Center (where the appropriate procedure is being followed). Compute the “Nearest Center” as number of moves needed to reach the Pokémon Center.
- c. The wild Pokémon *levels up*.
- d. **And the turn ends**

Outcome 2: If at the end of both attack phases, the Pokémon are still both alive, the dialog goes back to the main battle menu (with the three options: Fight, Switch Pokémon, and Run).

2. **Switch Active Pokémon:** if the player chooses to switch their *Active* fighting Pokémon, present the user with the names of the remaining Pokémon in their suite and ask them to choose one. The chosen Pokémon will become *Active* and will now represent the player in the current fight (if the player chooses to fight or escape next). Assuming the player has 4 other Pokémon in their party, here’s a sample output:



Pick another Active Pokémon (1, 2, 3, or 4):

- 1. Catterpie
- 2. Squirtle
- 3. Weedle
- 4. Blastoise

3

Your ACTIVE Pokémon is:

Name: Weedle (ACTIVE), HP: 40, A: 35, D: 30, S: 50, M: 20

What do you want to do (pick 1, 2, or 3):

- 1. Fight
- 2. Switch Active Pokémon
- 3. Run

3. **Run:** it is possible to escape from a battle with a wild Pokémon if you choose to **Run**. However, escape is not guaranteed; whether the player is successful at running away from the battle is determined by a calculation involving the speed of the combatants. When the escape is successful, the encounter ends immediately, and the player escapes to the nearest Pokémon Center. When the escape is not successful, the player will return to the battle menu (with the three options Fight, Switch Pokémon, and Run)

Escape is guaranteed if the wild Pokémon's Speed is strictly lower than the player's *Active* Pokémon. Otherwise, the chance of escaping is determined by the formula below:

$$F = \frac{A \times 32}{B} + 30 \times C$$

where:

A is the current Speed of the player's *Active* Pokémon, trying to flee

B is the opposing Pokémon's Speed divided by 4, mod 256, and

C is the number of times the player has tried to escape during the encounter (counting the current attempt).

Example: Pikachu wants to flee Weedle

Name: Weedle, HP: 40, A: 35, D: 30, S: 50, M: 20

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

A = 90 (the speed of the Pokémon trying to flee)

B = (50/4)mod 256 = 12.5 (based on the speed of Weedle, 50)

C = 1 (first attempt to flee)

F = (400/90) + 30 = 74

If F is greater than 255, the player escapes automatically. Otherwise, a random number is generated between 0 and 255. If that number is less than F, the player escapes. If not, the escape fails.

Note: since wild Pokémon are scattered at random locations throughout the land, there is a possibility for the player to encounter more than one wild Pokémon in one turn. You can resolve this situation as you see fit: you can battle just one Pokémon per turn, or battle all of them. You cannot, however, delete any wild Pokémon. Make sure to document your approach.

Summary of wild Pokémon encounter(s):

1. The player could **win** the battle, in which case they acquire the wild Pokémon.
2. If all the Pokémon in the player's party have fainted, the player will **lose** the battle, which will change their location. They will be teleported to the nearest **Pokémon Center**.

Pokémon Center procedure:

- all Pokémon will be healed to full HP
- the player is given the chance to reorganize their Pokémon suite. Present the user with the opportunity to select any Pokémon from their Pokédex into their suite. Remember, they cannot have more than 6 Pokémon in their suite/party.

- the starting location for the next turn is the location of the Pokémon Center.

Welcome to the Pokémon Center. All Pokémon in your party have been healed. Do you want to change the members of your party (Y/N): **Y**

Here are all the Pokémon in your Pokédex. Choose at most 6 numbers, followed by Q for quit:

1. Catterpie
2. Squirtle
3. Weedle
4. Blastoise
5. Bulbasaur
6. Charmander
7. Squirtle
8. Pikachu



3 4 5 6 7 8

You have chosen the 6 Pokémon below. Please choose which one should be the ACTIVE Pokémon:

1. Weedle
2. Blastoise
3. Bulbasaur
4. Charmander
5. Squirtle
6. Pikachu

1

Your ACTIVE Pokémon is:

Name: Weedle (ACTIVE), HP: 40, A: 35, D: 30, S: 50, M: 20

“Take charge of your destiny.”; Trainer Encounters

We first checked if the player was in proximity to other wild Pokémon. After all possible encounters with wild Pokémon are resolved, check if the player advanced on **a tile occupied by a Gym**. If they occupy the same tile, the player and the Trainer associated with that Gym will have an encounter.

If an encounter with a Trainer occurs, battle information should be displayed, with the player's *Active* Pokémon, the Trainer's *Active* Pokémon, their respective HP bars, and an option menu (2 choices, see below). At any time, the player may carry up to six Pokémon in his or her party; the *Active* Pokémon in automatically sent into battle. At the start of each battle-turn, the player can choose to **Fight** or **Switch** the *Active* Pokémon for another in their party. Fleeing from trainers is not an option!

You arrived at the Gym and the trainer is there. She wants to fight.
Her ACTIVE Pokémon is:

Name: Pikachu, HP: 35, A: 55, D: 30, S: 90, M: 50

Your ACTIVE Pokémon is:

Name: Weedle, HP: 40, A: 35, D: 30, S: 50, M: 20

What do you want to do (pick 1, 2, or 3):

1. Fight
2. Switch Active Pokémon

Each option leads to an action as follows:

- **Fight:** a fight with a Trainer is like a loop where every iteration is a battle between the *Active* Pokémon for each side. The same battle rules apply as in the wild Pokémon battles:
 - The opponent Pokémon cannot flee; it will always decide to attack
 - Compare the Speed values of the two battling Pokémon to see who attacks first
 - Run the first phase of the attack, followed by the second phase of the attack
 - if after one of the phases of the attack the player's Pokémon faints, go back to battle-menu (two options: Fight or Switch), which will allow the player the choice to send out another Pokémon to take its place, drawn from his or her party.
 - The Trainer's *Active* Pokémon *levels up*
 - if after one of the phases of the attack the Trainer's Pokémon faints, the next Pokémon in line will become the *Active* Pokémon for the Trainer.
 - The player's *Active* Pokémon *levels up*
 -
- **Switch Active Pokémon:** if the player chooses to switch their *Active* fighting Pokémon, present the user with the names of the remaining Pokémon in their suite and ask them to choose one. The chosen Pokémon will become *Active* and will now represent the player in the current fight.

Outcomes of the Trainer Encounters:

- A. All the Trainer's Pokémon faint. The player wins the battle.
 - a. The player acquires all the Pokémon in the trainer's party.
 - b. The player's *Active* Pokémon *levels up*
 - c. The player earns a **badge** from the Gym, **5 Pokéballs**, and **60 points**
 - d. If the player happens to be on the same tile as this particular Gym in the future, **the player will not be able to battle the Trainer again.**

- B. All the player's Pokémon faint. The Trainer wins the battle
 - a. The Trainer's *Active* Pokémon *levels up*
 - b. The player will be teleported to the nearest Pokémon Center, where procedure will be followed.

"I will show you that my love for my friends permeates every cell in my body."

Joining Pokémon Party, The Pokédex

When the player catches or defeats a wild Pokémon, the newly acquired Pokémon will automatically join the player's Pokédex. Out of all the Pokémon in the Pokédex, only 6 can be in the player's Active Party. These are the Pokémon that are sent into battle. The other Pokémon remain in the Pokédex; they can become part of the Active Party only when the player is at a Pokémon Center.

If the Active Party has less than 6 Pokémon, add the newly acquired Pokémon to the Active Party. If the Active Party already has 6 Pokémon, ask the player if they want to add the new Pokémon to their Active Party. If the player says **Yes**, ask the user to pick one Pokémon which will not be in the active party anymore.

Awesome! You acquired Charizard, who joins the Pokédex. Your battle party is full. Do you want to add Charizard to your Active Party?
(Y/N): **Y**

Which Pokémon should not be part of the Active Party anymore:

1. Weedle
2. Blastoise
3. Bulbasaur
4. Charmander
5. Squirtle
6. Pikachu

2

Great! The following Pokémon are now in your Active Party. Choose one of them to be your Active Pokémon:

1. Weedle
2. Charizard
3. Bulbasaur
4. Charmander
5. Squirtle
6. Pikachu



4

“Me give up? No way!”; Pokémon - Level Up!

When a Pokémon wins a battle and makes an opponent faint, the Pokémon gains experience and *Levels Up!* At the beginning of the game, each Pokémon starts in Level 1. When a Pokémon grows one Level, its stats will increase. For each level gained, the following stats will increase by 1/50 (or 2%) of the **base stat values**: HP, Attack, Defense, Speed. All values round up to the nearest integer.

Note: The Attack and Defense stat values cannot exceed the Max Attack/Defense value. If the base Max value is smaller than the base Attack (or Defense) value, then by leveling up the Pokémon will not be able to increase that stat, but it will be able to increase all other stats (see examples 2 and 3 below).

Example 1:

Venusaur - Level 1

HP: 80, A: 82, D: 83, S: 80, M: 100

Venusaur - Level 2

HP: 82, A: 84, D: 85, S: 82, M: 100

Venusaur - Level 3

HP: 84, A: 86, D: 87, S: 84, M: 100

Example 2:

Charmander - Level 1

HP: 39, A: **52**, D: 43, S: 65, M: **50**

Charmander - Level 2

HP: 40, A: **52**, D: 44, S: 67, M: **50**

Charmander - Level 3

HP: 41, A: **52**, D: 45, S: 69, M: **50**

Example 3:

Caterpie - Level 1

HP: 45, A: **30**, D: **35**, S: 45, M: **20**

Caterpie - Level 2

HP: 46, A: **30**, D: **35**, S: 46, M: **20**

Caterpie - Level 3

HP: 47, A: **30**, D: **35**, S: 47, M: **20**



“Do you always need a reason to help somebody?”; Random Events

At the end of every turn, after all encounters have been resolved, the following events can occur:

1. There is a 30% random chance that another wild Pokémon just spawned in the player's path. The player is presented with the option to catch it, or release it. If the player chooses to catch it, the wild Pokémon joins the party and the number of Pokéballs goes down by 1. If the player chooses to release it, the wild Pokémon disappears, just as suddenly as it showed up:

Suddenly, Charizard appears in your path. Do you want to:

1. Catch it
2. Release it

1

Awesome! Charizard joins the Pokédex. You still have room in your battle party. Do you want to make Charizard your Active Pokémon?
(Y/N): **Y**

2. There is a 25% random chance the player discovers a hidden treasure. There is a 50-50 chance the treasure can be
 - a. **Pokéballs:** Inform the player of the lucky event and award them 2 Pokéballs. Suppose you had 4 Pokéballs before the event, here's a sample display:

Great news! You have stumbled upon a hidden stash of Pokéballs. Your reserves increased to 4 Pokéballs.

- b. **A magic food that makes one Pokémon level up:** Inform the player of the lucky event and ask them to choose which Pokémon should level up

Great news! You have stumbled upon a hidden stash of Poffins. Your Pokémon are delighted. Choose one of them to feast on the Poffins and Level Up:

1. Bulbasaur
2. Pikachu

3. There is a 20% random chance that one of the Pokémon in the player's party will age out and die. Remove the Pokémon from the player's party.

Oh no! After years of legendary battles, Bulbasaur died of old age. Thank you for your companionship, dear friend.

“Me give up? No way!”

The game ends when:

1. The player chooses to end the game. In this case, the player **loses** the game.
2. The player earns 6 Badges. The player **wins**.
3. The player acquires Pokémon of 8 different types. The player **wins**.



Express your joy or disappointment with the outcome of the game and offer the player the choice to play again, from the beginning of the game. Write the name of the player and their total points accumulated in a file named *resultsPokemon.txt*, as outlined below.

“The important thing is not how long you live.

It’s what you accomplish with your life.”

For each game won, before starting a new game (or before ending the program), record the player name and performance (Won/Lost, followed by **number of points**) in a file named **resultsPokemon.txt**. If there are previous players recorded in the file, the current player’s performance should be appended at the end of the file. The output file should have *tabulated values* in the following format:

<Name>	<Won>	<points>
Sansa Stark	Y	75
Jon Snow	N	48

Note the *gorgeous* alignment of the names, Y/N and points. Ahhh... so satisfying!

Very important:

Think of the game you are developing as a real application. Try to anticipate player error and respond accordingly. For example, if the player can only choose between 4 menu options but

they choose option 5, your game should not crash, but instead display appropriate error messages and ask the player to choose again. Other examples of player error: trying to explore locations outside the map size or entering a letter instead of a number. In general, when testing your program, it is recommended that you try to “break” it. Fix all the instances that would cause errors or unwanted game outcomes, so that during the grading interview your program is unbreakable.

Extra credit (10 points total possible):

#1: (3 points) At the end of each game, present the player a sorted list of the top 5 players **who have won the game** and their stats: name, number of points.

#2: (3 points) Implement a new feature of the Pokémon game. In your PDF report, document your new variables or classes and how they influence the rules and the outcome of the game. You must significantly alter the flow of the game for your additional feature to be awarded the Extra Credit points.

#3: (4 points) present your project in a video, or in class (see below)

Minimum Requirements: Your implementation of the Project 3 should have:

- At least 4 user-defined classes
- At least 2 of these classes should have 4 or more data members.
- At least 1 class must include an array of objects from a class that you created.
- Appropriate methods for each class (including getters and setters)
- Your project implementation must include at least:
 - At least 6 **if** / **if-else** statements
 - At least 4 **while** loops
 - At least 4 **for** loops
 - At least 2 nested loops (these count in the 4 **while** and 4 **for** loops above)
 - At least 7 strings variables/data members
 - Reading from files (the provided text files or files of your own creation)
 - Writing to a file (`resultsPoke.txt`)
- Your project must have an interactive component (ask the player for input, create menus for choices, and so on). In other words, it's a *game*!
- You must display stats at each turn. It helps you debug, and it helps the user interact with your game in a more meaningful way.
- As part of the interactive component, you project must include a **2D map**; the characters involved in the project interact with one another and make decisions based on their environment. You must display the map at each turn.

Project timeline:

Checkpoint 1: Choose a Project (Pokemon or Create-Your-Own), meet with TA or CA before Wednesday, November 13

If you choose to implement the Pokemon project, you will need to come up with your own program structure. You will need to decide how the information will be stored in objects and how it will be passed between objects. You need to choose which classes to define, what are their data members, and which class/object is responsible for each part (functionality) of the game.

The choice of classes is entirely up to you. But to ensure you did not choose an approach that is not feasible, we require you meet for 15 minutes with the professor, a TA, or a CA, to go over your classes. **This meeting is mandatory.** You must choose a slot from the [Moodle scheduler](#) and have the meeting before Wednesday, November 13 at 6 PM. This meeting will count as interview grading in terms of the no-show and (re)scheduling policies. See the [syllabus](#) for more information.

Failure to schedule and attend the Project 3 Design Meeting will result in a 10-point penalty on your Project score.

Note: Even if you choose to create your own project, the design meeting is mandatory. Follow the instructions for the Project Proposal in the *ChooseYourOwn* Project write-up.

Checkpoint 2: Submit class files & Code Skeleton via Moodle, due Wednesday November 13 @ 11 PM

Your .h files should be *complete* with all the data members and member functions you will be using for each class. For the class implementation .cpp files, you should fully implement simple functions like your getters and setters. For more complex functions you can include function stubs with detailed comments. At a minimum, the input parameters, output type and pseudocode description should be present.

For example, if we were stubbing a function to implement bubble sort and return the number of swaps we might give in our code skeleton:

```
/*  
    1. Compare adjacent elements. If the first is greater than the  
       second, swap them.
```

2. Do this for each pair of adjacent elements, starting with the first two and ending with the last two. At this point the last element should be the greatest.
3. Repeat the steps for all elements except the last one.
4. Repeat again from the beginning for one less element each time, until there are no more pairs to compare.

```
*/
int bubble_sort(int arr[], int size)
{
    int swaps = 5;
    return swaps; // function returns expected type (int)
}
```

Your Code Skeleton should be inside your driver file and it should contain detailed comments with pseudocode explaining the functionality of the project.

You must submit your Class Files and Code Skeleton to Moodle to get full credit for the assignment. Failure to submit the Class Files and Code Skeleton will result in a 10-point penalty on your Project 3 score.

Checkpoint 3: Final Deliverables due Wednesday December 4 @ 11 PM

The final version of your project will be due on **Wednesday December 4 @ 11 PM** (no early submission bonus available and there will be no extensions). You must submit a .zip file to Moodle which includes:

- All .h and .cpp files, including the main driver program, correctly indented and commented.

Checkpoint 3A: Project Report - Reflection Activity, due Sunday December 8 @ 11 PM

Write a 1-2 page report containing answers to the following **Reflection questions**:

1. How did you prepare for the Project?
2. How did you develop your Code Skeleton? In what way(s) did you use your Code Skeleton?
3. Reflect on how you could have done better, or how you could have completed the project faster or more efficiently.
4. In addition, write a paragraph answering the following question, in the context of the Project in CSCI 1300:

Did you have any false starts, or begin down a path only to have to turn back when figuring out the strategy/algorithm for your Final Project program? Describe in detail what happened, for example, what specific decision led you to the false

starts, or, if not, why do you think your work had progressed so smoothly. In either case, give a specific example.

Note: all reflection papers must be individual.

Submit a PDF file of your report to the Moodle dropbox by **Sunday December 8 @ 11 PM**

Failure to submit the Report will result in a 10-point penalty on your Project score.

Checkpoint 4: Interview Grading - will begin on **Thursday December 5**, and must be completed at the latest on **Friday December 13** (reading day).

Checkpoint 5: Extra Credit Opportunity #3- 4 points (also for students choosing Pokemon) (due Sunday, December 8th @ 11 PM)

- Make a 5 minute (+ or - 1 min) video explaining:
 - The project idea
 - Implementation and approach
 - A demonstration of the working project
 - A **Google form** will be made available to **submit a link to your video**.
- **OR** you can present your project in lecture (5-7 min presentation).
 - You must **sign up to present by Wednesday, December 4 @ 11:55 PM**
 - Limited slots will be open and will be a first come, first serve
 - Presentation should include:
 - project idea,
 - implementation and approach, and
 - a demonstration of the working project.
- You **cannot** earn extra credit for doing both the video and the presentation in class

Collaboration:

All work for this assignment (and course in general) must be your own, original work. You may work together, but your assignment submission is **your own**. Your work may not include code taken from online resources like Chegg or StackExchange, or from other students (past or present), even with modification. Any such instances constitute Academic Dishonesty (passing off others' work as your own) and will earn you a 0 on the assignment and a trip to the Honor Code Council. If you aren't sure if something is okay, then please just ask!

Points:

Note: If your code does not compile, you cannot score above 40 points for the project.

20 points - meets minimum requirements specified on the first page (# of classes, loops, etc.)

40 points - interview grading

- TA's questions about your project
- algorithms descriptions, comments, good style
- code compiles
- if your code does NOT compile, you can get at most 20 points from meeting the minimum requirements, and at most 20 points from the interview

40 points - project functionality

- the game plays at outlined in the project description
- your solution accounts for user error

3-10 points extra credit

- (3 pts each) Extra features you can build into your game program
 - (4 pts) Create a 5 min (± 1 min) video explaining your project implementation & demo, **or**
 - sign-up to present and demonstrate your project in lecture (5-7 min presentation).
-

Possible Deductions:

- 10 points for not attending the project meeting (Checkpoint 1)
- 10 points for not submitting code skeleton and class files (Checkpoint 2)
- 10 points for not submitting report (Checkpoint 3A)