

1st Deliverable Submission Date: 1st May 2025 (11:59PM)

2nd Deliverable Submission Date: 12th May 2025 till 11:59 PM

FINAL PROJECT

Object Oriented Programming (CS1004) – Spring 2025

INSTRUCTIONS

- 1. Plagiarism in course project will result in F grade in the course.**
2. Each student will be working on the project individually.
3. The project has two deliverables
 - Deliverable 1: Class Diagram, clearly showing relationships between classes (Due 1st May)
 - Deliverable 2: Complete project (Due 12th May)
4. Combine all your work in one folder and compress it into a zip file. The folder must contain .cpp files and .h files (no binaries, no exe files etc.).
5. Each file that you submit must contain your name, student-id, and assignment # on top of the file in comments
6. Submit the solutions via google classroom. Submissions via email will not be accepted.
7. Use proper naming convention to name the file containing source code.
E.g. *A_i24xxx_project.cpp* , replace i24xxxx with your roll number and A with your Section.
8. Make sure you submit your project before the submission time. Late submissions won't be accepted even if they are late by just one minute.
9. You can earn bonus marks by implementing extra features in the project.
10. Use good programming practices (well commented and indented code; meaningful variable names, readable code etc.).
- 11. Follow the given instructions to the letter, failing to do so will result in a zero.**

Magical Pets Kingdom

1. Magical Pets (Dragons, Phoenixes, Unicorns, Griffins, etc.)

In this game, players can adopt different types of magical pets, each with its own unique abilities and characteristics. Players will have the opportunity to customize and develop their pets through training, battles, and strategic guild management.

Pet Types and Attributes

Dragons:

- Attributes: High health, high damage, and medium speed.
- Abilities:
 - Fireball Attack: Launches a fireball causing significant damage.
 - Dragon's Roar: Temporarily stuns nearby enemies.

Phoenixes:

- Attributes: High defense, medium health, and slow speed.
- Abilities:
 - Flame Healing: Heals the Phoenix or an ally over time.
 - Rebirth: Revives the Phoenix if it dies, usable once per match.

Unicorns:

- Attributes: High speed, low health, and moderate attack.
- Abilities:
 - Speed Burst: Increases speed for a limited time.
 - Magic Shield: Absorbs damage for a short duration.

Griffins:

- Attributes: Balanced in terms of health, speed, and attack power.
- Abilities:
 - Claw Strike: Powerful melee attack.
 - Dive Bomb: Aerial attack that stuns opponents.

Each pet has stats that can improve through leveling up, where the player can assign training points to enhance their pet's health, speed, attack power, or unlock new special abilities.

2. Pet Attributes and Abilities

Each pet has unique attributes and abilities that define how they interact with others and perform in battle. Players can strategically choose their pets based on their preferred playstyle and the needs of their team.

- Health (HP): Defines the amount of damage a pet can take before losing.
- Attack Power: Determines how much damage the pet deals during attacks.
- Speed: Affects how quickly the pet moves and reacts.
- Defense: Reduces the amount of damage taken from enemies.
- Abilities: These are special actions that can be used in combat, such as Fireball, Healing, or Speed Boost.

3. Guilds and Team Management

In this game, players can form guilds to manage and strategize their pets for team-based battles. Guilds serve as the to help players organize their pets into functional groups.

- **Guild Creation:** Players can create and manage their own guilds
- **Guild Structure:** Each guild can consist of multiple pets, and each pet can be assigned a role (e.g., tank, healer, damage dealer).
- **Guild Wars:** Guilds can engage in wars, where they face off against other guilds. Guilds will compete for rankings, rewards.

4. Training Camps

At Training Camps, players can improve their pets' attributes and abilities. These camps serve as development hubs for pets, where they can train to become stronger, faster, and more skilled for upcoming battles.

- **Training Mechanics:**
 - Players can send their pets to train and increase their stats (HP, Attack, Speed, etc.). The training process could include obstacle courses or target practice.
 - Training results in Training Points that players can use to improve specific attributes or unlock new abilities for their pets.

5. Arena Battles

The Arena is the heart of combat in the game. Players can battle their pets against other AI-controlled opponents in a variety of battle formats.

- **Battle Formats:**
 - 1v1: A simple duel between two pets.
 - 2v2: A team battle between two sets of pets.
 - Guild Wars: Larger-scale team battles involving multiple pets on each side.
- **Battle Mechanics:**
 - The goal of each battle is to either reach a certain number of points (e.g., 10) by defeating enemy pets or to eliminate all opponents.
 - Players can use abilities like Fireball, Healing, Speed Buffs, and Special Attacks to outsmart their opponents.
 - Arena Obstacles: Different arenas have unique obstacles (e.g., walls, traps, etc.) that players must navigate during combat.
- **Arena Strategy:**
 - Players can select their pets based on the battle type and create teams that work well together. For example, pairing a Phoenix with a Griffin to have healing and damage-dealing capabilities.

- Power-ups may appear randomly during the battle, providing temporary advantages (e.g., increased damage, defense, or health).

6. Shop and Items

The Shop offers various items that players can purchase to assist during battles or improve their pets' performance.

- Items Available:
 - Healing Potions: Restore a pet's health during battle.
 - Mana Potions: Refill the pet's ability to use special abilities.
 - Buff Items: Temporary boosts to attributes like speed or attack.
 - Shields: Reduce damage for a set period.
- Item Usage:
 - Players can purchase items using in-game currency earned from battles or achievements.
 - Items can be used at strategic points during the battle to either recover health, shield against damage, or boost an ability.

7. Saving and Loading Player Progress

To ensure players can continue where they left off, the game will have a save/load system.

- Data Storage:
 - Player and pet data will be saved in files (e.g., text files), storing essential information like player profile, pet stats, guild membership, and training progress.
 - Auto-save features will ensure that the game automatically saves progress at key moments (e.g., after a battle or after reaching a checkpoint).
- Load Progress:
 - Players can load their saved progress upon starting the game to resume from where they left off.

8. Handling Invalid Input and Game Exceptions

The game must handle invalid input and game exceptions to ensure smooth gameplay.

- Input Validation:
 - When players enter commands (e.g., pet names, item purchases, battle decisions), the game checks if the input is valid. Invalid inputs will prompt an error message and request corrected input.
- Exception Handling:
 - The game will catch exceptions like crashes or invalid file loads and display a user-friendly message, such as "An error occurred. Please try again." This will ensure that players don't lose their progress due to unforeseen errors.

9. User Interface and Game Flow

The user interface (UI) will be designed to provide a seamless experience for players. The UI will include elements like:

- Pet Selection Screen: Allows players to select their pet for battles.
- Guild Management Screen: Where players can organize their guilds, assign pets, and plan strategies.
- Training Camp Interface: A place for players to train their pets and assign attributes.
- Battle Arena: Displays the battle arena where combat takes place.
- Shop Interface: Where players can purchase items to aid in battles.

10. Bonus Features

- Adding sound
- Adding simple animations when a player is being hit
- Saving player progress, so that next time player with a same name plays 1 Player mode he restarts from the same level.
- Intelligent bots
- Canons fired can also reflect of the walls at angles other than 90

11. Starter Code

We have provided starter code in C++ to draw basic shapes which can be used for level generation and generating other game objects. You will change it and add functionality according to the project statement.

To execute the starter code on Ubuntu, you need to do the following:

- Extract the attached zip file.
- Open the terminal and navigate to the path of extracted directory
- Install the required libraries by executing the command below:
 - `bash install-libraries.sh`
- Compile the project by writing the command
 - `make`
- Run the main file
 - `./game`

Instruction to run the project in windows are in the separate file.

12. Important Note

Key Features You Must Use:

Abstraction, Inheritance, Polymorphism, Dynamic Memory Management, Operator Overloading.

It's a game but the emphasis is not on graphics, you should start by making a console-based game first and then move to graphics. Rubrics will be shared soon!