ID: 20239696

ALL source code available:

https://github.com/7LuxinC/programming_project

The Qt Zork Project details for Event-Driven programming module:

Major features:

C++ features:	Implementation: Brief
Wordle game	Wordle.h Wordle.cpp Wordledialog.h Wordledialog.cpp
Virtual functions	Virtual string getshortdescription()---zorkUL class + character
template classes	Division function –mainwindow.cpp
Multiple inheritance	
namespaces	Using namespace std; -- character class Using std::vector;
Programmer defined exceptions	listofWords function – wordle.cpp
Unions	
Operator overloading	Item *operator* + function --- character cpp
Abstract classes	Virtual getMove function -zorkUL
Copy constructors (deep, shallow copy)	
Bit structures	
Advanced pre-processor directives	#define MAX 1000 in wordle.h
C++ Object Construction Sequence	CreateRoom function in zorkUL
Initializer List	Character function –character .cpp
C++ References	addItems<with Room &room> function –character.cpp
pointers	Room *currentRoom in ZorkUL
Inheritance	Character + zorkUL
Destructors	~character function –character class
Memory management	**New** Room in CreateRoom in zorkUL
Header files (inclusion guards)	(#ifndef #endif) zorkUL.h --zorkClass
Arrays and pointers	Room *rooms[arrSize] -- zorkUL.h
Friendship	Friend Item operator function -- Item.h Room is friend class of character

ID: 20239696

Global variables	Int num = 0; mainwindow .cpp int counter = 0; wordledialog.cpp
Private, Public, Protected	zorkUL.h

***Some of the feature showed below from week 05 to week09

Features in QT for lab exercise:

Week 1

Install Qt and get the base zork code working

Week 2

1. See if you can add a new command word "Teleport" that moves you to another room (maybe a random room)
2. Make sure setup and upload your code to GITHUB your journey starts here!

Week 3

(*brief what you have to do in week3)**

1. Start a new project using the **QT Widgets Application** template.
2. Creat **Forms/mainwindow.ui** file.
3. Drag and drop a **Text Edit** and a **Push Button** to the application.
4. You can also add new components from your code, if you want. Open the **Headers/mainwindow.h** file.
5. Add function in push button.

Week 4

1. Rewrite your zork code
2. Create room teleport through an array of pointer

Week 5

Implement some of the C++ features we have covered in this weeks lectures into your Zork project

- Inline function ----------- getmoving function in character.cpp
- c++ references ---- addItems function – character.cpp
- default parameters ---character (int move = 0) in character.h
- function templates --- Division function –mainwindow.cpp

Week 7

Implement some of the C++ features we have covered in this week's lectures into your Zork project

- friend functions Friend Item operator function -- Item.h
- initializer list Character function –character .cpp

ID: 20239696

- static class members static int const arrSize = 9; in zorkUL.h
- inheritance character inherited ZorkUL

Week 8

Implement some of the C++ features we have covered in this weeks lectures into your Zork project

- copy constructors functions
- inheritance character inherited ZorkUL
- virtual functions -Virtual string getshortdescription()---zorkUL class + character class
- abstract classes and pure virtual functions getmove() in zorkul.h

Week 9

Implement some of the C++ features we have covered in this weeks lectures into your Zork project

- overloaded operators item operator+ function in character.cpp

---

Tips to open the application:

Open executable file -> cs4076_project.exe