Guided Practice: Pointers in C

OVERVIEW: Pointers -- stores the memory address of another value located in computer memory. Pointers are challenging but powerful programming construct. In this lesson we will learn the pointers basics, benefits of using pointers, how to pass by pointers, and how to implement pointers with structs.

LEARNING OBJECTIVES

Basic objectives (to be practiced *prior* to class):

- *Explain the reasons why we use pointers and how to pass by pointers*
- *Students will be able to explain the Pointer basics*
- *Students will be able to Implement pointers with structs*
- Students will be able to use Pointers with List

Advanced objectives (to be mastered *during and after* class):

- Apply pointer basics and passing by pointers to complete project1.
- Implement pointers using structs in for project2.

RESOURCES FOR LEARNING

The following are default suggestions for learning the material in this lesson. You may use these plus any other additional materials you can find.

- Online lessons on zybook: Read the section on “Pointers”.
- tutorial videos: Watch the following videos (total running time = 20 minutes, 18 seconds)
- Visual Studio: Implement projects by writing programs in visual studio compile, run and test the program with different data input values and check your result with other teams.

Projects: The projects for this lesson are found on the moodle assignment tab. Work on these projects in the group assigned to you. After finishing your program, compile, run and test the program with different data input values and check your result with other teams.

SUBMISSION INSTRUCTIONS: Once you finish your project, submit your work via moodle or canvas.

Embedded System Programming I (EE 2450)

Lesson Plan

Parvaneh Ghaforyfard

Dept. Electrical Engineering CSU

Lesson: Pointers in C

Timeframe: 180 minutes

Materials needed: Computer, copy of project description and requirements posted on Moodle, or Canvas. Visual studio software for programming, editing, compile, and running.

Objectives: List out the basic objectives tied to pre-class activities and the advanced objectives tied to in-class and post-class activities.

Basic:

1. *Explain the reasons why we use pointers and how to pass by pointers*
2. *Students will be able to explain the Pointer basics*
3. *Students will be able to Implement pointers with structs*
4. *Students will be able to use Pointers with List*

Advanced:

1. Apply pointer basics and passing by pointers to complete project1.

2. Implement pointers using structs in project 2.

Background:

Students are familiar with functions, passing variables by value and by reference prior to this lesson. Students find pointers a very challenging topic in programming. Pointers are very powerful programming construct. This lesson will illustrate one example's beneficial usage of pointers, namely pass by pointer function parameters.

Introduction to Lesson:

During the week before the class students will go through the assigned online lessons on zybook, tutorial videos that explain pointers and basics of pointers.
They will have to do the participation activities, quizzes, and challenge activities for each section.

Procedure [Time needed, include additional steps if needed].

Pre-Class Individual Space Activities and Resources: Outline the major steps for the preparatory activities and be sure to tie the steps to the basic learning objectives you have noted above. Note resources required for learner preparation.

Steps	Purpose	Estimated Time	Learning Objective
Step 1: Read chapter 9 from zybook. 9.1 Why pointers: Pass by pointer example Do the following activities **9.1.1: Pointer example: Without pass by pointer parameters.** **9.1.2: Pointer example: Pass by pointer parameters.** https://learn.zybooks.com/zybook/CALSTATELAEE2450GhaforyfardFall2018/chapter/9/section/1	Introduce students to pointers and why using pointers.	**20**	**#1 basics**
Step 2: 9.2 Pointer basics Do the following activities **9.2.1: Simple pointer example.** **9.2.2: Printing with pointers** **https://learn.zybooks.com/zybook/CALSTATELAEE2450GhaforyfardFall2018/chapter/9/section/1**	Students will learn the basics of pointers through examples	**20**	**#2 basics**

Step 3: **9.4 Pointers with structs** **Construct 9.4.1: Member access operator** **Challenge activity 9.4.2: Struct pointers.** **https://learn.zybooks.com/zybook/CALSTATELAEE2450GhaforyfardFall2018/chapter/9/section/1**	Students will learn accessing struct's member variables by first dereferencing a pointer	**20**	**#3 basic**
Step 4: **9.9 Why pointers: A list example** **9.9.1: Vector insert performance problem.** **9.9.1: Program illustrating how slow vector inserts and erases can be.** **https://learn.zybooks.com/zybook/CALSTATELAEE2450GhaforyfardFall2018/chapter/9/section/1**	Students will learn the most important use of pointers: to create a list of items such that an item can be efficiently inserted somewhere in the middle of the list, without the shifting of later items as required for a vector. To further elaborate on the need for pointers, this section describes another of many situations where pointers are useful.	**20**	**#4 basic**

In-Class Group Space Activities and Resources.

Steps	**Purpose**	**Estimated Time**	**Learning Objective**
Step 1: Answer any questions submitted before class.	Clear up any general confusion or misconceptions.	**5 minutes**	**All basic LOs**
Step 2: Working in group assigned try to solve the simple assigned project by applying pointer basics and passing by pointers. After finishing your program, compile, run and test the program with different data input values and check your result with other teams.	Have students apply the concepts they learned in the individual space to a practical case study.	**25 min**	**#1 advanced**
Step 3: Working in assigned group try to solve the assigned project by implementing pointers using structs. After finishing your program, compile, run and test the program with different data input values and check your result with other teams.	Have students apply the concepts they learned in the individual space to a practical case study.	**30 min**	**#2 advanced**

Evaluation:

Analysis. Students in this class are ultimately specializing in computers. This is their first programming class in C. The in-class group space activity is designed to allow students to apply the general concepts they learn in the pre-class individual space activities in a mode that they are comfortable working in. Students will develop teamwork skills including communication skills, collaboration skills, accountability, and leadership skills through the term project. In class projects should help them develop a professional work ethic where timeliness and collaboration at the conceptual level are encouraged but where collaboration at the implementation level should be limited.