SARC Supplemental Instruction with Ana Laranjeira

COP 3223 Final Exam Review

Thursday, April 30th, 11:30 pm – 1:30 pm

Location: Key West CD (218 CD) in the Student Union and Penji

Notes:

This is a final exam review made by a UCF student who has NOT seen your test.

This exam review was made based on the content that has been teached on Dr. Pfeil´s lectures. Please, if you want confirmation that a certain topic is going to be in your exam, talk with your professor. I hope this material helps you with your final review.

Answers will be posted after this session.

There is a reference sheet with a summary of some of the contents. Access it with your UCF EMAIL

Problem 1
Create a function called Reverse that takes in an integer array and its size and reverses it. (Milo, 2022)

Problem 2
Create a function called changeCase that takes in a string and its size. If the string is written in lower case letters, transform it to lower case. If it is written in upper case letters, change to lower case. Assume the initial string has consistent case (no capital letter, for instance).

Problem 3
Code trace this program:

```c
// Exam 2 Review
// Code tracing
// Ana Esther Laranjeira

#include <stdio.h>
#pragma warning(disable : 4996)

void func1(int a, int b, int c);
void func2(int* a, int* b, int* c);

int main(void) {

    int a=5, b=7, c=8;

    func1(a, b, c);

    printf("Values of a, b and c inside main\n");
    printf("a=%d | b=%d | c=%d\n", a, b, c);

    func2(&a, &b, &c);

    printf("Values of a, b and c inside main\n");
    printf("a=%d | b=%d | c=%d\n", a, b, c);

    return 0;
}
```

Problem 4
Create a function called “getName” that prompts the user for a string and allocates it dynamically to return a string (character pointer) with the exact size needed for it. (Pfeil, 2026)

Problem 5
A new coffee shop wants to have a digital expandable system for its products. For this reason, you were hired to create a program that uses jagged arrays to do it. The rows of the array indicate product category, and within each category, there can be a personalized number of columns that correspond to the product type. Each space in the spot of the array is going to record how many products of that type and category there are in the inventory. Your program should ask the user how many categories there are, how many types in each category, and the quantities of the products, and then print the array. See a sample output:

```
How many product CATEGORIES do you want to insert? 3
How many TYPES of products there are in category 1? 1
Enter the QUANTITY for category 1 product 1: 3


How many TYPES of products there are in category 2? 2
Enter the QUANTITY for category 2 product 1: 3
Enter the QUANTITY for category 2 product 2: 4


How many TYPES of products there are in category 3? 3
Enter the QUANTITY for category 3 product 1: 1
Enter the QUANTITY for category 3 product 2: 2
Enter the QUANTITY for category 3 product 3: 3


Final inventory
3
3 4
1 2 3
```

Problem 6

a) Let´s help a librarian. Create a structure “book” that includes a pointer to a character (name of the book), its release year (int), and the number of the last page read (int). Use typedef to be Book.

b) Create a function called constructBook. That takes in a file and returns a book structure. Your function should read the name of the book, its release year, and the last page read from the file. The file is in the txt format, and each line has the following format.

<AUTHOR> <YEAR> <PAGE>

Obs: assume the file has already been opened in reading mode in main.

c) Create a main to call the previously designed function and structure. In it, open the file “booklist.txt”, ask the user how many books there are, create a dynamic array with the exact number of books mentioned, use the function you wrote in the previous exercise to build each book by using the file information, and allocate each book to the dynamic array.

d) Create a function called “oldestBook” that takes in the dynamic array with all books and prints the name of the oldest one.

e) Create a function called “updateLastPage” that takes in a pointer to a book, prompts the user for the new page they stopped at, and updates that information in the book structure.

f) How would you modify the originally created book structure so that it becomes a linked list?

Problem 7

a) Design a structure to register the employees of a company. It should contain their ID (integer), age (integer), and name (string that does not exceed 100 characters).

b) Create a function called “create_employee” that creates an employee, by taking in an ID, an age, and a name, and returning a structure of the type determined on the last question.

c) Create a function called update “ID” which takes in a pointer to the employee struct and lets the user update the ID of that specific person.

IF WE HAVE EXTRA TIME OR IF YOU WANT TO PRACTICE MORE

Problem 8

Write a function that returns the smallest number in an integer array. Write a function that takes in the array address and its size. Use pointer tracing.

Problem 9

Write a function that calculates the range of an array by implementing the function previously defined. The range is the difference between the largest and the smallest number in a list. (Guha, Arup)

Problem 10

Create a program that declares an array of 10 integers using dynamic memory allocation

Problem 11

What is the output of this code?

```c
#include <stdio.h>

int main(void) {
    int a = 5, b = 6, c = 7, d;

    d = a++ + --b * 5 - ++c / 2 + a * b % c;

    printf("a = %d, b = %d, c = %d, d = %d\n", a, b, c, d);

    return 0;
}
```

Problem 12

Write a program that reads a file named words.txt, which contains an arbitrary number of words (one per line). Each word has no whitespace and is at most 50 characters long. The program must create a new file named words_info.txt. For each word from the input file, write the word followed by its length, separated by a single space. Preserve the original order of the words. (Paredes, Yancy slides)