

**Lab 9: Pointers**

**CSE 4108**

**Structured Programming I Lab**

November 2023

# 

# 

**Lab Tasks**

**Recursion:**

1. **With Great Power, Comes Great Responsibility!:**

***power.c*** (Can be found on Chapter 9, page 205) can be made faster by having it calculate in a different way. We first notice that if n is a power of 2, then can be computed by squaring. For example, is the square of , so can be computed using only two multiplications instead of three. As it happens, this technique can be used even when n is not a power of 2. If n is even, we use the formula = . If n is odd, then = . Write a recursive function that computes .

(The recursion ends when n = 0, in which case the function returns 1).

To test your function, write a program that asks the user to enter values for x and n, calls power to compute , and

then displays the value returned by the function.

2. **GCD:**

Write a program in C to find GCD of two numbers using recursion.

3. **Binary to Decimal:**

Write a program in C to convert a binary number into decimal using recursion.

**Pointers:**

4. **Pointers:**

Write a program that declares and initializes (to any value you like) a double, an int, and a char. Next declare and initialize a pointer to each of the three variables. Your program should then print the address of, and value stored in, and the memory size (in bytes) of each of the six variables. Here’s an example of the desired output:

The address of char c is 0x61feff

The address of int i is 0x61fef4

The address of double d is 0x61fee8

The address of char\* pc is 0x61fef8

The address of int\* pi is 0x61fef0

The address of double\* pd is 0x61fee4

The value of char c is a

The value of int i is 1

The value of double d is 2.500000

The value of char\* pc is 6422271

The value of int\* pi is 6422260

The value of double\* pd is 6422248

The size of char is 1

The size of int is 4

The size of double is 8

The size of char\* is 4

The size of int\* is 4

The size of double\* is 4

**Note:** Use the “0x%x” format specifier to print the addresses in hexadecimal.

**Hint:** Use the function sizeof() to get the memory size.

5. **Swap**:

Write a program that contains the following function:

**void swap (int \*x, int \*y);**

The function will swap the values pointed by x and y. The main function will pass addresses of two variables which need to be swapped.

6. **Coin Change:**

In **Lab 1 - C Fundamentals** (*Problem: Counting Denominations*), you wrote a program that takes the U.S. dollar amount as input and then shows how to pay that amount using the smallest number of $20, $10, $5, and $1 bills. Modify the program so that it includes the following function:

**int pay\_amount (int dollars , int \*denominations, int n);**

Here, denominations is an array containing n different denomination amounts that will be taken from the user. The function determines the smallest total number of all denominations necessary to pay the amount represented by the dollars parameter.