CS-UY 2214 — Recitation 1

Introduction

Complete the following 33 exercises. Put your answers in a plain text file named `recitation1.txt`. Number your solution to each question. When you finish, submit your file on Gradescope. Then, in order to receive credit, you must ask your TA to check your work. Your work should be completed and checked during the recitation session.

Please note that your solutions must be in a plain text file. Other formats, such as PDF, RTF, and Microsoft Word, will not be accepted. Here are some recommended editors that produce plain text files:

- Notepad (comes with Windows)
- TextEdit (comes with Mac OS); note that if you are using TextEdit, you need to select "Make Plain Text" from the Format menu before saving the file
- gedit (available on most Linux distributions)
- nano (available on most Linux distributions)
- Sublime Text
- VSCode
- Atom
- Vim
- Emacs

For questions that require a solution expressed as an image, submit the image as a separate file. The image file should be named `recitationnnqm`, where n is the recitation number and m is the question number; use an appropriate suffix (either jpg or png).

Problems

1. Consider the following circuit. (Diagram would be included as recitation1q1.jpg or .png)

   Express the circuit as a Boolean expression, using only AND, OR, and NOT. Provide a truth table for this circuit.

2. Using only AND, OR, and NOT gates, construct a circuit diagram that will calculate XOR. Your circuit will have two inputs A and B, and one output Y. Submit your answer as an image, in accordance with the instructions at the beginning of this document. Your image may be a diagram created in a drawing program, or it may be a photograph of a hand-drawn diagram on paper. (Image would be included as recitation1q2.jpg or .png)

3. Convert the following decimal numbers into binary.

   - (a) 27
   - (b) 105

4. Convert the following binary numbers into decimal.

   - (a) 1110
   - (b) 0101

5. Convert the following decimal numbers into hexadecimal.

   - (a) 255
   - (b) 128

6. Convert the following hexadecimal numbers into binary.

   - (a) 1a2b
   - (b) cafe

7. Convert the following decimal numbers into 8-bit binary 2's complement. Write all the digits.

   - (a) 67
   - (b) 67

8. Convert the following 8-bit binary unsigned numbers into decimal.

   - (a) 01101100
   - (b) 10000001
   - (c) 00000000

9. Convert the following 8-bit binary 2's complement numbers into decimal.

   - (a) 01101100

- (b) 10000001
- (c) 11111111

10. Consider the following C++ program. Do not run it.

```
# include < iostream >
using namespace std ;
int main () {
int i = 10;
int count = 0;
while ( i > 0) {
i -= 2;
count ++;
}
cout << " Completed " << count << " iterations " << endl ;
return 0;
}
```

    Your friend Rufus argues that this program has an infinite loop, and will therefore never end. Is he right or wrong? Justify your opinion with a persuasive explanation, but do not type in or run the program. Predict the output of the program.

11. Design a circuit using only NAND gates that implements a half adder. Show the circuit diagram and the truth table. (Image would be included as recitation1q10.jpg or .png)

12. Explain the difference between combinational and sequential logic circuits. Provide examples of each.

13. What are the advantages and disadvantages of using binary, decimal, and hexadecimal number systems in computer science?

14. Describe the process of converting a decimal number to its two's complement representation. Illustrate with an example.

15. Explain overflow and underflow in the context of binary arithmetic. Give examples of each.

16. What is the purpose of a truth table? How is it constructed?

17. Simplify the Boolean expression: (A + B)(A' + C) using Boolean algebra.

18. Draw a Karnaugh map for a function with three inputs (A, B, C) and output F = (0, 2, 4, 6). Simplify the expression using the map.

19. What is a multiplexer (MUX)? Explain its functionality and provide a diagram of a 4-to-1 MUX. (Image would be included as recitation1q18.jpg or .png)

20. What is a demultiplexer (DEMUX)? Explain its functionality and provide a diagram of a 1-to-4 DEMUX. (Image would be included as recitation1q19.jpg or .png)

21. Design a circuit using only NOR gates that implements an OR gate.

22. Design a circuit using only NAND gates that implements an AND gate.

23. Explain the concept of De Morgan's Law and provide examples of its application.

24. Convert the following binary numbers into Gray code:

    - (a) 101101
    - (b) 001110

25. Convert the following Gray code numbers into binary:

    - (a) 110110
    - (b) 011001

26. Write a C++ program that calculates the factorial of a given integer. The program should handle invalid input (negative numbers) gracefully.

27. Write a Python program that finds the largest number in a list of numbers.

28. Write a Python function that checks if a given string is a palindrome.

29. Write a C++ function that reverses a given string.

30. Explain the difference between a while loop and a for loop in C++. Provide examples of each.

31. Explain the concept of recursion in programming. Provide a simple example of a recursive function.

32. Consider the following C++ program. Do not run it. What will be printed?

```
#include <iostream>

using namespace std;

int main() {
  int x = 5;
  int y = 10;
  int z = x + y;
  cout << "The sum of x and y is: " << z << endl;
  return 0;
}
```

33. Take a screenshot of your Linux terminal showing the successful compilation and execution of a simple "Hello, world!" program. (Image would be included as recitation1q32.jpg or .png)

34. Take a screenshot of your Linux environment showing a file manager with at least three files or directories visible. (Image would be included as recitation1q33.jpg or .png)