Student Number ____________________ Name ____

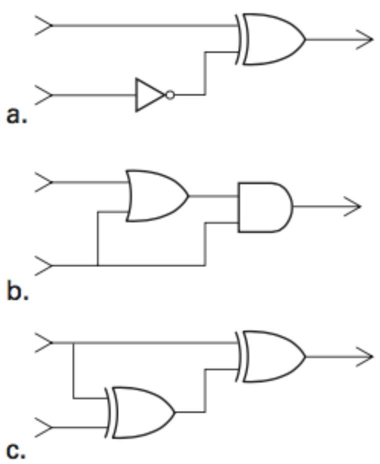
CENG105 Introduction to Computer Engineering 22.11.2017
Fall 2017
Midterm

1) (10 pts) Write an **algorithm** that will describe how a web browser works when the user enters a web address (URL) and the corresponding web page is displayed on the browser. Please describe all steps explaining how DNS is used and IP address resolution is done.

Algorithm accessWebPage(URL)

a. Ask DNS the IP number of the URL
b. If a valid IP number is received, then
    i. Use the IP number to get the HTML page using POST command from the web server.
    ii. If the HTML page is received from the web server then
        1. Display the HTML page on the browser.
    iii. Else
        1. Display an error message ("Page does not exist!")
c. Else
    i. Display an error message on the browser ("URL does not exist")

2) (10 pts) For the three circuits below, write a table for each showing the output for all possible input value combinations. Consider that all circuits have two input bits and one output bit.



Input1	**Input2**	**Output**
0	0	1
0	1	0
1	0	0
1	1	1

Input1	**Input2**	**Output**
0	0	0
0	1	1
1	0	0
1	1	1

Input1	**Input2**	**Output**
0	0	0
0	1	1
1	0	0
1	1	1

3) (5 pts) What is the equivalent **binary form** for the hexadecimal number D39F5C?

110100111001111101011100

4) (5 pts) Convert the binary number **110011** to its **base 10** equivalent (show steps):

32 + 16 + 2 + 1 = 51

5) (5 pts) Convert the integer base 10 number **541** to its equivalent **binary form** (show steps):

541/2 = 1
270/2 = 0
135/2 = 1
67/2 = 1
33/2 = 1
16/2 = 0
8/2 = 0
4/2 = 0
2/2 = 0
1/2 = 1
541 = **1000011101**

6) (5 pts) Express **5¾** in binary notation.

1001.11 (4+1 . 1/2+1/4)

7) (5 pts) Perform the following additions in binary notation:

```
a.  01111
   +00001
    10000

b. 1010.110
   + 11.101
   1110.011
```

8) (10 pts) Represent **-27** (negative twenty seven) in *two's complement notation* using 8 bits.

base10 (27) = *base2* (00011011) = *2sComplement*(11100101)

9) (10 pts) When a CPU executes a program, it keeps performing a three-step process known as the machine cycle. The three steps are Fetch, Decode, and Execute. Please describe the operations of these steps in a machine cycle.

a) *Fetch* – copies the instruction in the memory cells addressed by the Program Counter to the Instruction Register and increment the Program Counter to point to the next instruction in main memory.
b) (b) *Decode* – decodes the bit pattern in the instruction register to determine the operation required and generates control signals to setup the computation circuits.
c) (c) *Execute* – performs the operation specified by the instruction in the instruction register (and write back the result to the destination register or memory cell).

10) (15 pts) Answer the following:

a. How many bits an IP address has? (for IPv4)
32 bits

b. Show the IP address **95.183.128.255** in binary form.
01011111.10110111.10000000.11111111

c. How many different IP addresses can be used in the (v4) IP address range from **95.183.128.0** to **95.183.128.255** (Çankaya University IP range). Show your calculation steps.

255 – 0 + 1 = 256 (0 to 255)

d. How many IP addresses do we need in the world with the following restrictions? Show your calculations clearly.
   i. Half of the population in the world aged 10 and above has a cell phone,
   ii. 10% of population has a electric car with an IP,
   iii. 40% of the population has a laptop, tablet, or PC at home,
   iv. 20% of the population has a laptop or PC at office.

   Word population is 7 billion
   If average life expectancy is 70 years, population of 10-70 is ~**6 billion**
   Cell phone count is ~**3 billion**
   Electric car count is ~**700K**
   Laptop/tablet/PC@home count is ~**2.8 billion**
   Laptop/PC@office count is ~**1.4 billion**
   TOTAL IP count = 6 + 3 + 0.7 + 2.8 + 1.4 = **13.9 billion**

11) (10 pts) What is the output of the following *python* program:

```
num = 100
while (True):
      if (num == 0):
            break
      print(num)
      num = int(num / 2)
```

```
100
50
25
12
6
3
1
```

12) (10 pts) Turkish alphabet consists of 29 characters (a to z). And you want to use upper and lower characters (like *a* and *A*). How many bits *at least* do you need to encode all Turkish characters in binary form? Show a sample encoding table for all those characters of your design (characters vs. bit encoding).

All chars # (a-z, A-Z) = 29 x 2 = 58

Codes from 0 to 57

57/2 = 1
28/2 = 0
14/2 = 0
7/2 = 1
3/2 = 1
1/2 = 1

We need at least 6 bits to represent codes from 0 to 57

Char	**Code**
a	000000
b	000001
c	000010
ç	000011
…	…

z	011100
A	011101
B	011110
C	011111
Ç	100000
….	….
Z	111001