1. [2 points] State 2 examples for each of the following algorithm types. So, all in all: 4 examples. Examples cited in the course slides are not counted.

Unsolvable:
•
•

Undecidable:
•
•

2. [1 point] State two differences between machine code and assembly language code.
•
•

3. For the following binary number: 10110
   a. [1] What is the decimal equivalent if we interpret it as unsigned int?
b. [1] What is the decimal equivalent if we interpret it as signed int?

c. [2] Take the number you calculated in “b” above and present it in IEEE 754 binary presentation of floating point single-precision.

d. [1] Why 2’s complement is signed number presentation used in almost all computers?


b. [1] Can we design the control unit without knowing the instruction set? Justify.

c. [1] Can we design the control unit without knowing the high-level language that will be used to write programs on that computer? Justify
5. Given the following logic circuit:

![Logic Circuit Diagram](image)

a. [1] What is the logic expression (i.e. Boolean expression) that represents the above circuit?

b. [1] What are all the values of a and c for which the value of f is 1?

c. [1] Take only the AND gate with a, b, and c as input from the above figure. Substitute that gate with one or more NAND gates, yet give the same result.

6. [1] Why do we need both ebp and esp to manage the stack in IA-32 ISA?
7. [1] After the following piece of C code, what will be printed?
   ```c
   int a = 0x20, b = 02;
   a >>= b;
   printf("a=%d, b=%d\n", a, b);
   ```

8. [2] Implement, in C, the function whose definition is shown below (assume m and n are integers larger or equal to 0):
   
   
   ```c
   A(m, n) = \begin{cases} 
   n + 1 & \text{if } m = 0 \\
   A(m - 1, 1) & \text{if } m > 0 \text{ and } n = 0 \\
   A(m - 1, A(m, n - 1)) & \text{if } m > 0 \text{ and } n > 0 
   \end{cases}
   ```

9. [1] In algorithm analysis, we mostly care about large values of n and worst-case scenario. Why is that?

10. [1] List a scenario where executing an \( O(n) \) algorithm can be faster than \( O(\log n) \) algorithm even if the same machine and same input are used.