

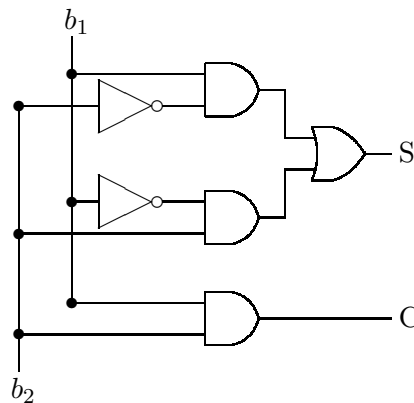
Review of the first 15 lectures

In contrast to quizzes, the exams are closed book. Calculators or laptop computers are not allowed. Good examples of questions are quizzes and homework assigned at the end of each lecture. The material breaks down in three parts:

1. computer science concepts: algorithms, computer architecture, functionality of operating systems, mass storage, formal languages, flowcharts and pseudocode, logic gates, transistors and flip flops, adder circuits, functional programming.
2. programming in Python: variables and assignments, elementary and composite types, intrinsic operations, if else elif, while and for, queues and stacks, top down design, functions, lambda.
3. mathematical CS: computer algebra, binary representations of numbers, precision and accuracy, Maple's types and control structures, boolean algebra, truth tables, simulation and Monte Carlo, histograms.

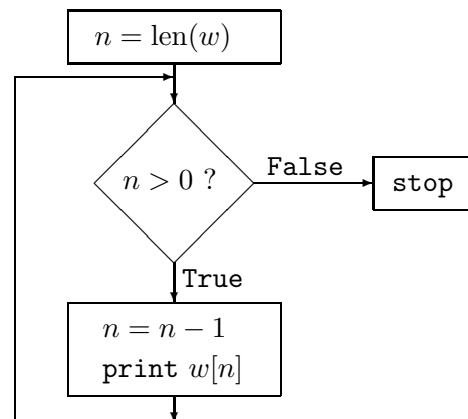
This sheet contains some preliminary examples of questions which may help you prepare for the midterm exam.

1. List the characteristics of an algorithm.
2. How many bytes are 15 terabytes?
3. Explain the difference between $34/87$ and $34.0/87$.
4. What is an interrupt?
5. Write a Python program that takes on input a nonnegative number less than 100. The number represents the number of cents. Compute the minimal number of coins (quarters, nickels, dimes, and pennies) that are needed to obtain the value of the input number. For example, for 68 cents we need 2 quarters, 1 dime, 1 nickel, and 3 pennies.
6. Consider the circuit shown below. For $b_1 = 1$ and $b_2 = 0$ as input, indicate on the picture below the input and output for every logical gate.



7. Draw the flowchart for the algorithm to search for the maximum in a list of unsorted numbers.

8. An exclusive or, denoted as XOR returns False only when the inputs are both the same, and True otherwise.
 - (a) Give the truth table for the exclusive or.
 - (b) Show how the XOR can be realized with NOT, OR, and AND by giving the logical expressions and their corresponding truth tables.
 - (c) Draw the circuit for the XOR, using the symbols for the gates NOT, OR, and AND.
9. Give the Python code to generate a list of all tuples containing all possible input values for a Boolean expression, i.e.: [(False,False),(False,True),(True,False),(True,True)]. Extend the code so it will work for triplets.
10. Write a program to sum the values of coins. The coins are represented by a sequence of characters, like `pnqdpqnppdqddpp`, where `p`, `n`, `d`, `q` represent respectively a penny (\$0.01), nickel (\$0.05), dime (\$0.10), and quarter (\$0.25). Use a dictionary.
11. Consider the flowchart, that takes on input the word w :



- (a) Explain what the algorithm defined by the flowchart does.
 - (b) Use a `while` to implement the algorithm in Python.
 - (c) Use a `for` to implement the algorithm in Python.
 - (d) Modify the Python code so that, instead of `print`, it will create a new string that will be printed after the loop has terminated.
12. Explain how to simulate an unfair coin that when flipped returns head with probability 0.6. Give a Python function for such unfair coin.
 13. Define a Python function that returns as a tuple the roots of a polynomial $p(x) = ax^2 + bx + c$, using $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$. Write the function so that 1 is the default value for a .

Please note the policy on skipping exams: If an exam is missed, then greater weight will be placed on the final exam, especially on the material covered on the missing exam. **What this means is** that if you decide not to take one midterm exam, your final exam will be weighted for one hundred points more. **What it does NOT mean is** that you can drop the score of a midterm exam. If you take the midterm, then your score counts. So, please be prepared when you show up for the exam.