Summary Analysis

1.1

1. Computation is a general term for any type of information processing. It is a process following a well-defined model that is understood and can be expressed in an algorithm, protocol, network topology, etc.
2. Data is defined as a representation of facts, concepts, or instructions in a formalized manner, suitable for communication, interpretation, or processing by a human or a computer.
3. Information is the processed data that is i) accurate and timely, ii) specific and organized for a purpose, iii) presented within a context that gives it meaning and relevance, and iv) can lead to an increase in understanding and decrease in uncertainty.
4. Information science is a scientific field that studies information systems and information-related activities such as information: analysis, collection, classification, storage, transfer, manipulation, dissemination and protection.
5. Informatics is an applied form of information science.
6. The key characteristics of information systems are its abilities to acquire, transform and exchange information.
7. The key characteristic of information systems are the base requirements of any technological advancement.
8. Computer science is a set of skills and technologies focusing on development, programming and use of computers. Computer science deals with all practical activities related to information.
9. The fundamental operations of every computer are its ability to: i) input information, ii) store information, iii) process information, and iv) output information.
10. The key difference between a fixed-program computer and a stored-program computer is that the first one is designed to do only very specific things, and that changing its program would require rewriting, restructuring or redesigning the machine. On the other hand, a stored-program computer stores (and manipulates) a sequence of instructions, and has a set of elements that will execute any instruction in that sequence. By treating those instructions in the same way as data, a stored-program machine can easily change the program.
11. The elements of von Neumann model include: i) input, ii) output, and the computer, consisting of iii) memory and iv) central processing unit, which contains: iv-a) control unit and iv-b) arithmetic logic unit.
12. Hardware - physical, tangible parts or components of computer.
13. Software – a set of instructions, data or programs used to operate computer and execute specific tasks.
14. System software is designed to run a computer’s hardware and application programs. Application software is a collection of programs used by end users that help them perform a specific task.
15. Analog signal are continuous while digital signals are always in “steps”.
16. Digitization is the process of converting analog to digital signals (information).
17. Using digitization we prevent the loss of information. As a consequence, more space and time is required to store and transmit the same information.
18. Humans natively store and manipulate information in analog form.
19. Computers natively store and manipulate information in digital form.
20. The analog form is native for humans and as such will always be present. The digital form is likely to dominate in the future. Analog used to be more accurate, consumed less space and could be transmitted faster than the digital form, but with improvement of the technology that difference diminished. The digital information is resistant to noise and can be copied without quality degradation.

1.2

1. An algorithm is a step by step method of solving a problem in finite number of steps.
2. Cooking recipe, to-do list are some of the examples of using algorithms in non-technical world.
3. Computer programs are the realization of algorithms on the computer. Programmers “tell” the computer what to do and how to do it.
4. There are many ways to represent an algorithm. Some or more and some less formal. Flowcharts are among the most used.
5. According to the way the steps are executed, we categorize the algorithms into: linear, branching, cyclic. Typical algorithms are combination of all three.
6. In this lesson we covered basic mathematical operators for addition (+), subtraction (-) multiplication (\*), power (\*\*) and division operators.
7. For division we used float division(/), floor division (//) and modulo (%).

2.1

1. In our history, we have used non-positional and positional numeral systems to represent numerical values.
2. In a non-positional numeral system, each symbol represents the same value regardless of its position. The numerical value is determined through the addition of values.
3. We (humans) natively use the decimal system. The main reason is the number of fingers on our hands, 5 + 5 = 10. That is why early symbols were all values divisible by 5 and 10.
4. The decimal positional numeral system (Hindu-Arabic, or Indo-Arabic) uses 10 symbols and the value of each symbol is determined by its face and place value. Thanks to positional numeral systems, mankind could easily express large numbers without the need for addition (and subtraction).
5. Zero has a twofold purpose: as placeholder and no value.
6. The base is the number of different symbols – digits (and letters) that a system of counting is using to represent numbers.
7. Machines cannot understand the notion of numbers. At best they can differentiate between two states: circuit open and circuit closed. That is equivalent to base 2 or the binary numeral system.
8. The hexadecimal and octal systems are used because binary numbers are not always practical (the sequence can get very long). The base for hexadecimal is 16 and for octal is 8. Each hexadecimal numeral corresponds to 4 binary bits, and each octal numeral corresponds to 3 binary bits.
9. Conversion from and to decimal numbers is covered in slides 37 – 43.

2.2

1. Variable is a memory location in a computer that contains certain value. The purpose of variable is to store and label information.
2. To change the value of the variable, you have to assign a new value either explicitly or through an expression.
3. So far, we mentioned three variable types: numerical variables, strings and logical variables. In our examples we mainly used numerical variables and occasionally strings.
4. To read from the standard input (console) we use input() function. It returns string type.
5. To convert from string to number we use int() function, and to convert number to string we use str() function.
6. Linear algorithms are algorithms where every step is executed once and only once.
7. There are many ways to swap values between two variables, but the easiest is to add another variable to temporary store the value of the first and then perform the swap.
8. To extract digits from the number you need to use floor division, modulo operation and know the base of that number.

3.1

1. Logic is the study of the criteria used in evaluating inferences or arguments.
2. An inference is a process of reasoning in which a new belief is formed on the basis of or in virtue of evidence or proof supposedly provided by other beliefs.
3. An argument is a collection of statements or propositions, some of which are intended to provide support or evidence in favor of one of the others.
4. Statements are the kind of sentences that are either true or false.
5. Sentences that do not have a truth value are non-statements.
6. In contrast to mathematical algebra (dealing with numbers), the focus of Boolan algebra is on applying mathematical laws to logical statements. Boole’s approach allowed truth to be systematically and formally proven through logic equations.
7. The values of Boolean variables are TRUE and FALSE.
8. There are three fundamental Boolean operators: negation - NOT (~ or ¬), conjunction - AND (∧), disjunction - OR (∨).

3.2

1. Boolean variables in Python can have only True and False values. The correct syntax is a word True / False, starting with a capital letter.
2. The order of evaluating Boolean operators in Python is: 1) not, 2) and, 3) or.
3. To ensure that (a section of) and expression is evaluated first, we can use parenthesis.
4. Comparison operators are operators used to compare two values. The result of comparison is a Boolean value.
5. ‘=’ is assignment operator, while ‘==’ is comparison operator.
6. If statement contains keyword ‘if’, Boolean expression (condition) and the body of the statement (could be one ore multiple lines). If the expression is evaluated to be True, lines from the body will be executer. Otherwise they will be ignored.
7. Else in if … else … statement denotes the beginning of the part of the code which will be executed only if the expression is evaluated to NOT True (= False).
8. Branching algorithms are the type of algorithms that contain steps that will be either executed ONCE or NEVER

4.1

1. De Morgan's Laws describe how mathematical statements and concepts are related through their opposites. In propositional logic, De Morgan's Laws relate conjunctions and disjunctions of propositions through negation.
2. Deductive reasoning is a top-down logic approach, that links premises with conclusions. It is the process of reasoning from one or more statements to reach a logically certain conclusion.
3. To reach a truthful conclusion in deductive reasoning, all premises must be true, terms clear, and rules of deductive logic followed.
4. Tautology is a formula or assertion that is true in every possible interpretation.
5. Contradiction is an unsatisfiable statement, both through negation or affirmation.
6. Contingency is a formula that is neither Tautology nor Contradiction.
7. An argument is valid, if it is impossible for its premises to be true, while its conclusion is false (the conclusion must be true, if the premises are true).
8. An argument is sound if it is valid and the premises are actually true.
9. Validity can be proven using truth tables.
10. In this course we showed three rules of inference: 1) modus ponens (the law of detachment), 2) modus Tollens (the law of contrapositive) and 3) hypothetical syllogism (chain argument).

4.2

1. Similar to branching algorithms, cyclic algorithms have condition and body. If the condition is evaluated as true, the steps in the body will be executed. If not, the steps in the body will be skipped. However, once the steps in the body have been executed, the algorithm will “jump back” to the condition again and perform the evaluation.
2. Steps in cyclic algorithm can be executed once, multiple times or never.
3. One iteration is the execution of all steps in the body of the cyclic algorithm.
4. Cycles where the condition is checked after the body can have 1 or more iterations. Cycles where the condition is checked before the body can have 0 or more iterations.
5. While loop is a control flow statement. As long as the condition is satisfied, the steps in the body of while loop will be executed repeatedly.
6. In almost every case, we have to initialize some variables before entering the loop.
7. If the algorithm is forever stuck in the cycle, we call it infinite / dead loop.

5.1

1. Set is well defined collection of objects. We say “well defined” because it does not vary from one person to another, “collection” as it means group, and “objects” as any entity, number, or item we want to include. Set is a group of all objects with some specific property.
2. Sets are typically represented by capital letters. The members of a set are given inside of curly brackets (braces), separated by commas (‘,’).
3. ‘∈’ is a sign of relation, and is read as “element of”. It is used to show if an element is a member of given set.
4. A set is container for objects and as such it doesn’t have order, nor can it have doubles.
5. Cardinality is the size of the set. It shows how many members that set has. The notation for cardinality (for example set A) is n(A) or |A|.
6. Sets are equal if they precisely have the same members.
7. A set A is a subset (‘ ⊆ ’)of another set B if all elements of the set A are elements of the set B.
8. A set is a proper subset of another set if and only if every element in the first set is also in second set, and there exists at least one element in the second set that is not in first set.
9. A universal set ( ξ ) is the set containing all objects or elements and of which all other sets are subsets.
10. The empty set ( ∅ ) is the unique set that has no elements. Its size or cardinality is zero.
11. A Venn diagram - is a diagram that shows all possible logical relations between a finite collection of different sets.
12. Union ( ∪ ) of two given sets is the smallest set which contains all the elements of both sets.
13. Intersection ( ∩ )of two given sets is the largest set which contains all the elements that are common to both sets.
14. Two sets are disjoint sets if their intersection is the empty set.
15. Set difference of two sets A and B is the largest subset of set A such that none of its elements are members of set B.
16. Complement of a set is the set of all elements of universal set ξ which are not elements of that set.
17. Set-builder notation is a way to describe a set by saying what properties its members have. For example: all integers less or equal to 10 {x ∈ Z | x ≤ 10 }.
18. Interval are all numbers between two given numbers. We distinguish between open intervals ‘(’ and ‘)’ and closed intervals ‘[’ and ‘]’.
19. Power set is a set of ALL subsets of a set. For a set S with n elements, the cardinality of its power set is 2n elements.
20. The Cartesian product (or cross product) of two sets A and B, (A × B ) is the set of all possible ordered pairs where the elements of A are first and the elements of B are second.
21. A Cartesian coordinate system is a coordinate system that specifies each point uniquely in a plane by a set of numerical coordinates, which are the signed distances to the point from two fixed perpendicular oriented lines, measured in the same unit of length. Each reference line is called a coordinate axis or just axis (plural axes) of the system, and the point where they meet is its origin, at ordered pair (0, 0). The coordinates can also be defined as the positions of the perpendicular projections of the point onto the two axes, expressed as signed distances from the origin.

5.2

1. Sets can be defined using curly braces { }, and typing set members inside, separated with comma (,) sign. Alternatively, set() can be used to create an empty set.
2. To add element to the set, use it’s variable plus the method .add(), writing a new member inside parenthesis.
3. Elements from the set can be removed using remove() or discard() method. Both will remove an element if it exists in the set, but if not, remove() will raise an error, while discard() will just ignore it.
4. len() command will return the cardinality of a set, or the number of elements in tuple, number of characters in string, etc.
5. To test if an element belongs to the set we use **in** operator (e.g. a ∈ A).
6. FOR loop is used to iterate over a sequence.
7. FOR loop repeats one block of steps fixed number of times.
8. An iterator is a variable that will be assigned one element from the sequence with each iteration. Typically, the first iterator is named ‘i’, then ‘j’, ‘k’, …
9. range() function is used to generate a sequence of numbers over time.
10. The parameters of range function are start, stop and step. Only stop is required, while start and step can be omitted. If omitted, start = 0 and step = 1.
11. In WHILE loop statements in the body are executed as long as the given condition is evaluated as true. In FOR loop statements in the body are executed fixed amount of times, corresponding exactly to the length of the sequence given. Every for loop can be written as while loop, but not all while loops can be written as for loops.
12. String is an ordered, immutable sequence of characters.
13. Tuple is an ordered, immutable sequence of objects.
14. Index refers to a position within an ordered sequence. Using index, we can access each element in a sequence directly.

6.1 venn diagrams

1. Categorical proposition is a statement about the relationship between categories. It consists of a subject and a predicate, and show quality and quantity.
2. There are four standard forms:
   1. A – all S are P (universal affirmative)
   2. E – no S is P (universal negative)
   3. I – some S is P (particular affirmative)
   4. O – some S is not P (particular negative)

6.2

1. List is a mutable ordered sequence of elements.
2. Each value contained inside a list is called an item or a member.
3. List can be created as an empty list using list() command, or by listing all values, separated by commas between square brackets.
4. The main benefits of using lists are: keeping data together, ability to perform same methods and operations on multiple values at once, and condensed code.
5. Elements in list are accessed using indices, either positive (starting from the beginning) or negative (starting from the end).
6. Elements in list are counted using len() function.
7. Operator in is used to test membership of an element in a list.
8. The methods to add element to a list are .append() and .insert(). Append will add the element to the end, while insert will insert the element at the given position.
9. There are three options to remove an element from a list. Command del() can be used, or methods .remove() or .pop().
10. Slicing is a quick way of retrieving a subset of items from a list. Inside of curly brackets, you define start and end index separated by colon, and optionally step value.
11. Operator + is used for list concatenation, while operator \* is used for list repetition.
12. A List comprehension is a syntactic construct in Python that allows to create a list in a single line of code.
13. The required part of list comprehension is the expression, followed by for clause. Optionally, there can be additional for clause(s) and one or more if clause(s) at the end.
14. The main difference between assignment and copying lists in Python is that by assigning multiple variables to the same list, makes the change in one reflect on all the others. Copying, in general sense, creates a new object, so that the changes in the copy do not influence the original.