1)The goal of the book is to teach us to think like a computer scientist.

2)The most important skill of a computer scientist is problem solving.

3)The differences between high-level languages and low-level languages are:

* It is much easier to program in a high-level languages
* Programs written in a high-level language take less time to write, they are shorter and easier to read
* High-level languages are portable
* Low-level programs can run on only one kind of computer and have to be rewritten to run on another

Examples:

High-level: Python, Java C and C++

Low-level: Machine languages and assembly languages

4)An **Interpreter** reads a high-level program and executes it, meaning that it does what the program says. It processes the program a little at a time alternately readings lines and performing computations.

A **Compiler** reads the program and translates it completely before the program starts running.

5)A **program** is a sequence of instructions that specifies how to perform a computation.

The **computation** might be something mathematical, such as solving a system of equations or finding the roots of a polynomial, but it can also be a symbolic computation such as searching or replacing text in a document or compiling a program.

6)**Bugs** are programming errors.

**Debugging** is the process of tracking down and correct the bugs.

7)a-A sentence must begin with a capital letter and end with a period.

b-We can read the poetry of e. e. cummings without spewing error messages.

c-Because if there is a single syntax error anywhere in your program, Python will print on error message and quite it, and you will no be able to run your program

8)**Runtime error** is also called exception because it usually indicate that something exceptional and bad has happened.

9)**Semantic error** is the third type of error. The program you wrote is not the program you wanted to write.

10)Because programming is the process of gradually debugging a program until it does what you want.

11)English is a natural language because people speak it. The English was not designed by people, they evolved naturally.

**Python** is a formal language.

12)One of the two basic rules is pertaining to tokens and structure. Tokens are the basic elements of the language, such as words, number and chemical elements. The other type of syntax error pertains to the structure of a statement.

13)**Parsing** is to figure out what the structure of the sentence is.

Humans and computers need to parsing because once they have parsed a sentence, they can figure out what it means, or the semantics of the sentence.

14)

|  |  |  |
| --- | --- | --- |
|  | NATURAL LANGUAGES | FORMAL LANGUAGES |
| AMBIGUITY | People deal with it by using contextual clues and other information. | Designated to be nearly or completely unambiguous, which means that any statements has exactly one meaning, regardless of context. |
| REDUNDANCY | Employ lots of redundancy. | Less redundant and more concise. |
| LITERALNESS | Full of idiom and metaphor. | Mean exactly what they say. |

15)The best way of reading a program in distinction to reading in English according to the authors is identify the tokens and interpret the structure.

**Grammar**

1)a-The main verb tense is Simple Present.

Yes, there is other verb tense; Simple Future.

b-The goal of the book.

-We will learn to program.

-The single most skill for a computer scientist is problem solving.

-Problem solving means the ability to formulate problems, think creatively about solutions and express a solutions clearly and accurately.

The process of learning to program is an excellent opportunity to practice problem solving skills.

c-**Examples**:

Like mathematicians, computer scientists use formal languages to denote ideas.

Like engineers, they design things, assembling components into systems ans evaluating tradeoffs among alternatives.

Like scientists, they observe the behavior of complex systems, form hypotheses, and test predictions.

**Less relevant information:** On one level, you will be learning to program, a useful skill by itself. On another level, you will use programming as a means to an end. As we go along, that end will be become clearer.

d-

|  |  |  |  |
| --- | --- | --- | --- |
| VERBS | NOUS | ADJECTIVES | ADVERBS |
| Teach  Combines  Use  Denote  Designate  Assembling  Evaluating  Observe  Solving  Formulate  Think  Express  Turns out  Learning  Program  Become  Called  Practice  Become | Goal  Book  Scientist computer  Features  Mathematics  Engineering  Languages  Ideas  Systems  Behavior  Hypotheses  Test  Problems  Process  Chapter  Level | Complex (6)  Single (8)  Excellent (11)  Clearer (15)  Important (8) | Clearly (10)  Accurately (10)  Creatively (9) |

2)Addition: And (3-23), Also (3-20).

Contrast: Or (2-20), Although (7-9), However (5-25).

Examples: For Example (7-4), Like (1-4)

Reason and Cause: As (1-14), Because (2-22), Due to (2-12)

Results or effects: So (3-24), As a result (7-16)

Listing and sequencing: First (7-34), Finally (8-3), Second (2-8)

3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ABILITY | PROBABILITY | OBLIGATION | ADVICE | NECESSITY |
| Will be (1-16) |  | Have to (2-2) | Should (3-24) | Must (4-24) |
| Can (2-21) | Might (1-17) |  |  |  |
| Will not be able (4-29) | May (3-19) |  |  |  |
| Can not (6-28) |  |  |  |  |
|  |  |  |  |  |

4)-Low-level languages are used only for a few specialized applications.

-Python is considerated an interpreted languages because Python programs

are executed by an interpreter.

5)-If there is a single syntax error anywhere in your program, Python will print an error message and quit it, and you will not be able to run your problem. (Type 1)

-If your hypotheses was wrong, you have to come up with a new one. (Type 0)

6)-Meaning that it does **what** the program says. (2-16)

-That may be a little vague, but we will come back to this topic later **when** we talk about algorithms. (4-13)

-….**Which** means that any statement has exactly one meaning, regardless or context.(7-13)

-People **who** grow up speaking Natural Languages often have a hard time adjusting to formal languages.(7-21)

7)-This error are also called exceptions because they usually indicated that something exceptional has happened. (Present Perfect Simple. 5-3)

-When you have eliminated the impossible, whatever remains, however improbable, must be the truth. (Present Perfect Simple. 5-24)

-Programming languages are formal languages that have been designed to express computations. (Present Perfect Simple. 6-16)

-Python is an example of a high-level language; other high-level languages you might have heard of are C, C++, Pearl and Java.