**Software Videos With Questions**

MIT Introduction to programming: <https://www.youtube.com/watch?v=k6U-i4gXkLM>

Question 1.1: What are the skills which the lecturer says they want to develop in this course.

Question 1.2: The lecturer says that he will not distribute class notes. WHY does he say that they will not distribute class notes?

Question 1.3: At 16:21 the lecturer begins to talk about thinking like a computer scientist. Does he encourage memorization?

 Question 1.4: Distinguish between declarative knowledge and imperative knowledge.

Stop at time code 21:48

How to get a square root of x

G = guess

If G2  is close enough to x then stop

Else new guess G = (G+(x/G))/2

Go back to the test

Question 1.5: What is a fixed program computer?

Question 1.6: What is an interpreter?

Question 1.7: What is a “stored program computer”?

Question 1.8: What are the parts inside a stored program computer?

Stop at time code 31:17

Question 1.9: Does the lecturer say that some problems can be solved only in one language and not another?

Question 1.10: What does the lecturer say about “the best programming language”?

Question 1.11: What is the difference between a low level and a high level language?

Question 1.12: Distinguish between an interpreted and a compiled language.

Stop at 38:13

Question 1.13: What is syntax (as applied to a computer language)?

Question 1.14: What is “static semantics”?

Question 1.15: What is “semantics” as applied to a program?

Question 1.16: What does he mean when he speaks of “style”?

End at 44:20

MIT Introduction Lecture 2: <https://www.youtube.com/watch?v=Pij6J0HsYFA> (50:48)

Question 2.1: What are the three types of data the lecturer describes?

Question 2.2: What does the term “type conversion” mean?

Question 2.3: Why is type checking important?

Question 2.4: What does “order of precedence” mean?

Stop at 15:36

MIT Lecture 3: <https://www.youtube.com/watch?v=X6ilT3uUOBo> (50:59)

Question 3.1: What are some operations which are used on data?

Question 3.2: List 4 types of commands

Question 3.3: Give some examples of good programming style

Stop at 4:44

MIT Lecture 4: <https://www.youtube.com/watch?v=X6ilT3uUOBo> (51:26)

(Abstraction and functions)

Question 4.1: What is the purpose of a function?

Question 4.2: What is the purpose of the parameter inside the ( ) ?

Question 4.3: What is the purpose of “return”

[The computer language used here is Python, not C. Therefore some of the details will be foreign to you. Do not worry about that.]

Stop at 15:20

The discussion of global and local variables (bindings) is a very important idea.

Stop at 19:38

Intro to computer programming (motivation)

 <https://www.youtube.com/watch?v=OWsyrnOBsJs> (9:34)

Question 5.1: What is programming?

Question 5.2: Computers are (very smart / very stupid)

Question 5.3: According to the move, programmers work (mostly in isolation / mostly cooperatively)

Good introduction (computer – binary - programs – C)

<https://www.youtube.com/watch?v=AWliApDc61w> (17.22)

Question 6.1: A computer is nothing itself without programs (True/False)

Question 6.2: What is the distinction between system programs and application programs?

Question 6.3: What is a program?

Question 6.4: Why do computers use binary rather than decimal numbers?

Stop at 5:40

Question 6.5: What does CPU stand for?

Question 6.6: The machine actually understands what language? (Machine language)

Question 6.7: What is assembly language?

Question 6.8: What is an assembler?

Stop at 12:14

Question 6.9: What is a high level language?

Question 6.10: What is the name of the first high level language?

Question 6.11: What is the difference between source code and machine code?

Question 6.12: What are some advantages of learning C?

VERYQuick intro to C <https://www.youtube.com/watch?v=nXvy5900m3M> (23:56)

(Like drinking from a fire hose…but might be good as a review).

Computer science vs. Computer Engineering <https://www.youtube.com/watch?v=kHFjNPrmZRM> (7:32)

Intro to EECS <https://www.youtube.com/watch?v=3S4cNfl0YF0> he clicks and clears his throat (1:17:34)

Question 7. 1. Electrical Engineers design complex systems which are very reliable. (T/F)

Question 7.2. How can we get a good system on the first try? (By building and testing a model)

Question 7.3. How do you learn a language? (A) Learn all rules first (B)Practice, Theory, Practice (C) Experiment first then learn the rules

Question 7.4. PCAP means (Primitives, Combinations, Abstraction, Patterns)

End at 12:20

Question 7.5. What needs to be added to the bad plan for robotic steering of a car?

Question 7.6. Let’s say you are to the right of your path. Would it be helpful to know how far to the right you are? Why, or why not?

End at 22:00

 Variables and add two numbers <https://www.youtube.com/watch?v=SBQwQRwkg6U> (8:03)