1. C: I imagine that it is a piece of the code with all the variables piled under.

2. I: [...] what do you think about what an object is?
J: What an object is. When I have discussed and so on then I have said that an object is a class or can be a class, or can be a method in a class. Eh, when I have been studying I have tried to think of if you have a programme, I don't remember but? which consists of a couple of objects and the objects can be different classes which contain methods. In that way I tried to get a picture of what an object is, because it is easier for me to think that if I have a class and in the class we have a lot of objects.
I: Although now you drew it the other way around ...
J: Yes because it was like that I thought about it when I read the book, so I don't know really which is right. It's different when you ask different persons, yeah but, class is that an object. Yes it is an object but a method was also an object. Is it the class which contains different objects or is it an object that contains different classes. So that I don't know.

3. L: (Hm) I would probably almost describe it as a box where you put different characteristics, different things that this object will do.
[..]
L: It was then when we were supposed to do our own classes. Then I realize that it was so to speak just a storage space for methods that belong to certain objects.

4. G: [...] I remember so to speak that it was... if you will compare Basic with Java then it was so that large programmes in Basic became very unstructured, it's difficult to find how you will put it forward and structure it... run so to speak with classes, divide it up in different files so to speak and this whole part makes it so it can be built upon each other in some way, build so to speak programmes on other.

5. C: Then the class should really be something "clever" which contains that you shield, this is a class and in a class you could put everything under the same class although it's not very clever to do so if you want to use some things in other programmes. Then it is good to have them kind of shielded from each other. But the class is really just some blurred collection of, this I think belongs together, in some way.

6. I: What do you think is the point with having objects and classes?
B: That I don't know, what else would it be so to speak, if you didn't have it. It's necessary, isn't it, so to speak.

7. A: Oh, the point is that you have a pattern from the beginning, then you can make them green and blue if you want that and if you don't want it then you take them away and make them orange instead for instance. It is, it is about like a mathematical formula before you have put in the numbers. With this formula you can do plenty of different things and you can perhaps change an m and take it away and, well, you can do very much without affecting itself... if I want something to be 18, then I put in the numbers that makes it 18. It's the same here. If I so to speak want something to look exactly so.

8. E: But well, you divide the programme simply and then... theoretically you can always write everything in the same programme, or? Although it would be so incredibly... I don't know, it wouldn't work. (giggle)
I: Why do you think it wouldn't work?
E: You have to have everything in order, the structure, in the beginning you don't think so much about the structure but when you start programming some more then you realize how important it is [...]

9. C: Yes an object, you can have a rather physical image of it....
I: What did you say, physical?
C: Kind of, you can think of a car and then it has one variable for how many wheels it has, one variable for the size of the engine like that.

10. I: ...what do you think is the point with having classes and objects? Why do you create classes and objects?
N: It is because the programmes require it. That's the way it goes to create a programme. I don't know how to do it otherwise.

11. H: A class is well, yes, like I think a class is like a little programme, that's how I think of it, a small programme inside the whole big programme being kind of the main programme. Then a class is like a small programme which does certain things.

12. L: It was then when we were making our own classes. At that point I came up with that it was just kind of a storage space for methods that belong to certain objects.

13. O: [...] Eh, when you write a class, for instance class vector which we have had as a class particle, then you write well, yes, to be able to create an object of that class later you write how you want it to look like and that is how I see a class, that you will be able to create an object and some of what you will be able to do with this object in the different methods [...]

14. I: What do you think is the point of having objects and classes then?
K: The point is that... (pause)... well it is that you can write programmes easy for that purpose which you are looking for kind of.

15. K: But simply that it's easier with classes to get what you want from the programme or to write...
I: To write the programme?
K: Yes or to solve the assignment perhaps you can say. To write the programme can be too hard even with classes but, yes. To solve the assignment. If you take for instance assignment no. four, if you didn't have classes in it then it would have been really hard for sure.

16. I: What do you think is the point with having classes and objects?
C: Well, the point is really that you can have this clear image, this is how it looks in the real world. Yes but then it is something similar on the computer then. So that it can be described so to speak, therefore a rather clear image of it. Now I have a particle and I have a box and the particle has vectors. Yes it has it in the reality too in some way.
I: Precisely.
C: So you get a very concrete picture of also how you perhaps can put forward the programme if you will do something which in reality has a couple of objects, then they create one object at the time so to speak and then you link them together so they will behave as you want. So it is a very clear picture...

17. E: Class. Mm, it took a while before you came to grips with classes, what it really was actually, that I don't know if I still have. But classes which only contain a lot of methods for instance that you later use or, like how a vector works, it's a class for instance and, a bunch of computing vectors which you then will be able to call and use so that you don't have to write everything at the same place, a sort of classification of chapters or something similar.
I: Classification of... ?
E: But well, you divide the programme simply and then... theoretically you can always write everything in the same programme, or? Although it would be so incredibly... I don't know, it wouldn't work.

18. D: [...]But, no, in some way it feels like there are a lot you're expected to know and then a lot you're expected just to understand immediately and to put in the right context. Yes.
I: Is it the conception, what's on your mind then, what do you mean with that it's a lot?
D: It's so difficult, it feels like everything is just floating like this (laughter), ehm. No it is, what is it about. Therefore, oh...

19. M: the object is a kind of, what is doing something [..] because it is all about that something is going to happen.

20. F: (sigh) Object, it's hardly that I even come to any real grips with it. [...] I think you got rather bad conceptions of it anyway, this object oriented. You heard the word all the time but I find it very hard to put my finger on what it is and what's the difference compared to other programming languages. [...] No but they say, it says if you read and everybody says that Java is an object oriented programming language. But it doesn't tell me very much actually.

21. I: But this about class, I mentioned, how do you think about class?
C: That is a bit more diffuse actually. Class, it is that I would probably think of that a class contains, can contain a couple of objects or just one object and different operations that you can do in an object or between objects. So that you can also think of what it would mean in reality.
I: Okay.
C: Well, you can have a working space and a human that works there, then you have two objects and then they can so to speak interact with one another through different operations so to speak, what do I know, the human gets some coffee and then the coffee variable goes down at the working place and so on.
I: Okay (laughter). And what do you think the class now, now I want to...
C: Then the class would really be something smart containing what is to be shielded, so this is a class and a class you could put everything under the same class although it is not very clever to do if you want to use some things in other programmes. Then it is good to have them so to speak shielded from each other. But the class is just some blurred collection of, this I think belongs together, in some way.

22. I: [...] Eh, do you think it has been difficult to understand this?
F: Yes I do. It easily becomes very abstract. So it is very difficult to get a grip on it. Which is which and how it works, yes, for instance classes and objects and what it actually is and what it actually comes out to be everything. It turns out to be a lot, difficult to get a real understanding of it.

23. I: [...]. Do you think it has been difficult to understand objects and classes?
L: The class is no problem but it's the way I mix all the conceptions together, so I don't know if I have had that clear image of objects and there were no one who really used the word object .
I: Not on the lectures?
L:No I don't think so. [...] once in a while sometimes but not referred to them so to speak as they are called, particles and vectors, so to speak, never said really that it's an object. Possibly he said it in the beginning, yes this is what an object is, then I didn't really think he used the word.
I: So when you have written some code and so, it has not emerged that right here ...
L: No.
I:. .. creates an object or something?

L: No, I don't think so.
24. M: How I think about a class... well, like a ginger cookie cutter maybe. [...] It is more like a cast form then cause how you make new ones from the beginning identical one maybe, if you now will think about that. But which can be different very quickly. It can be... well, it is...
I: They can be different, what did you say?
M: They have the same origin in some way but they don't need to be identical because they come out from the same.
I: What is it that you think so to speak is different?
M: Different contents then, more like instance variables. They... well, you can do the same operations with them, you can do, with the same class you can in principle do the same elements and so on with all objects. If they are not too...
I: But element that is to say.
M: You can, if you have your cat object you can let the cat run even though it has three legs defined or two legs.
I: Yes, that's right.
M: You can still let it do certain stuffs anyway.
I: What do you think is then... the difference so to speak of class and object?
M: Class and object. Yes, that the class is the pattern over how the objects of the class look like. That's how I think of it.

25. F: [...] I think you use Java the way you learned it and programme but still it's difficult to return to basic and what the actual object is, that is the central idea and so on.

26. K: If you think of the Java program, that it is built of different objects and it is the objects we modify so that we can get what we want from it.

27. N: It is because the programmes require it. That's the way it goes to create a program. I don't know how to do it otherwise.

28. I: Do you find it difficult to understand classes and objects?
B: Yes I think so. It is very difficult to know if you kind of have understood it correctly because you could as well have understood quite wrong things. I don't know if I have understood it correctly.
:
29. I: How would you explain what an object is to a friend?
N: I'd just say that it is a part of the program.