

Read the whole question before choosing which question to do! Often the first parts of one question may look very nice but the question as a whole may be very nasty!

Read the damn question and make sure to do exactly what you are asked to do (ie Q1b you did not make a curried function despite being asked to make a curried function).

Get used to re-using previous sections of the question in subsequent parts.

Questions almost always ask you to reuse parts of previous questions in later parts. If you learn to use this then you will have a much easier time in exams than otherwise. This is especially true with FOCS.

Note that you are never asked ever to make a program that is efficient – you should never go out of your way to make anything even remotely more efficient. As long as your code is not ie exponential where linear would suffice.

Before choosing which question to do; you should read the entire question (ie parts a - f). Often questions which initially look easy have a nast bit at the end of the question.

As long as the strategy that your code tries to use is correct, exact syntax is relatively irrelevant.

Improve in Object-Oriented-Programming – this is a simple programming paradigm that it is possible to do very well in. Do this by using Java.

Design patterns are very common – learn them and how to implement them (compositve and decorator for example).

Learn Collections better – not just how to use them but also when to use them. The only way to do this is by using Java.

For example in OOP there is a question where all the work can be delegated to a TreeSet and you do almost no work at all. You must learn collections better – and java in general.

To have a `abc.def()` class you should have an `abc` package with a `def` class inside it – this lost you a mark in the exam.

Learn exceptions fully – they're not hard but they are useful and a very common topic.

In Java sometimes it is smart to extend collections and override some classes. However only sometimes. Nobody did it – John just said it was something you could do and would have been smart in one specific case.

Dealing with edge cases is crucial – case in point the Ceasar Cypher question – you did not handle spaces and assumed every single character was a letter!

Questions are likely to focus on non-overlapping topics. This means that you do not *have* to be brutally good on everything. If you use the ability to choose questions well then you can improve your marks significantly!

Exam questions are more likely to focus on things at the end of the course than at the beginning – the material at the beginning of the course is usually foundational and so there is not much to ask on it – hence you should ensure you know it fully (otherwise it will be hard to get good answers without a strong foundation) however often the meat of the questions will focus on the latter parts of the course.