Advantages and Disadvantages of Decomposition

- You have to really understand the problem you're trying to solve to be able to decompose it properly. Otherwise you'll end up writing code you don't need.

- I can break the problem I have to solve into smaller chunks and deal with them one at a time. This makes it much easier to deal with a complex problem.

- Decomposition is when I break up a program into small sections.

- Different people can code the sections of decomposed program at the same time. This makes writing a complex program quicker as the jobs can be divided up more easily.

- Teams have to work together really effectively when they are working on their own sections of a program. Otherwise the parts won't fit together properly.

- Once I have a useful function written I can probably reuse it in lots of different projects. This saves me loads of work.

- I find that using decomposition means that my programs often end up being shorter on the whole.

- If I write a function I can test it really well to make sure it always gives me the right result. It's much simpler to test small sections of code than a long program.

- If subroutines or functions have been documented well it makes maintaining program in the future much easier as they are already in smaller chunks.

- The sections of a program are sometimes called subroutines or functions.