What are the main differences between projects Debug and Release mode?

What is intellisense?

How can you locate code belonging to a given c++ statement?

Explain preprocessing, compilation, and linking phases

What are the advantages of defining separate header and source files?

Lesson 2

Describe the most important basic types in C and compare them with other languages you know

What is a type magnitude?

What is a type qualifier? Why would you need it?

Does C have support for complex numbers (you may need to research into this..)

Lesson 3

What are the kinds of errors that can occur when working with variables (e.g. overflow, domain error)? How would you replicate these errors in a test program?

What would the criteria for using float instead of double (or vice versa) in computational finance?

Remember that float has less precision than double. Do you want accuracy over speed, etc.

Compare the efficiency of ++i with that of i++

Describe the conversion scenarios between int and double, in particular data/type conversions

What is the value of d? Why?

double d = ½;

cout<<d;

Lesson 4

Compare C’s support for decision making with that from any other language(s) you know

What is the essential difference between while and do while? When would you use each one?

What happens to variables that are defined in a block? i.e., the variable lifecycle. Are they defined outside the block?

Is goto so bad after all? Is it better/worse than while/do while when implementing nonlinear iterative solvers?

Lesson 5

Explain *call by value* in relation to function call arguments; what is the possible performance impact? What are the advantages?

Could be if you copy a large object

Why are calls to recursive functions slower than calls to non-recursive functions in general?

What is meant by stack overflow?

Discuss the impact of global, static, and register variables on code maintainability; are they harmful?