1. Framework Fundamentals
   1. Managed vs non managed
      1. What is a managed framework and what is a non-managed framework?
      2. How is memory allocated and accessed in a non-managed framework?
      3. Is the .net framework a managed framework?
      4. Please describe what is the .net framework and its characteristics?
      5. Does C++ run in a managed environment?
      6. What’s the principal characteristic of a managed framework?
      7. What are the advantages and disadvantages of a managed framework?
      8. How code is built in the .net framework and how is executed?
      9. Does the .net framework binary code generated at compilation is native code?
      10. In Java, code is generated in byte code and then interpreted by the JVM, what is the name of the intermediate code generated in the .net framework?
      11. What is the common language runtime (CLR) in .net framework?
      12. Please describe up to 3 languages you can use to develop applications in .net framework
      13. Why is possible you can develop a library in F# and you can use it in a C# project?
      14. Do you consider the .net framework as a successful development platform and what could be the reason of its success?
      15. If speed is critical for one application or service, would you consider its implementation in the .net framework or you better go the native way?
      16. If security is critical for one application or service, would you consider its implementation in the .net framework or you better go the native way or choose another platform?
      17. Does the .net framework run in embedded systems?
      18. Please enumerate up to four types of visual studio projects you can create targeting the .net framework?
      19. How do you think development will be done in 30 years?
   2. Object's life cycle
      1. What’s the life cycle of an object in the .net framework?
   3. Garbage collection
      1. What’s the function of the garbage collection in the .net framework?
      2. Does the garbage collection process run automatically, what are the conditions that trigger this process?
      3. Can I force the execution of the garbage collection process?
      4. If my class uses resources that need to be disposed when my instance is no longer needed, what interface does it need to implement?
   4. Base objects/root object
      1. What class is the root class of all the .net classes?
      2. What is boxing/unboxing/casting in .net?
   5. Base object methods
      1. Please enumerate up to 3 public methods of the Object class?
      2. What methods can I override from the Object class?
      3. What is the meaning of overriding a method?
      4. What is the meaning of overloading a method?
      5. How do you define optional parameters in a method?
      6. If you have multiple optional parameters in a method, how can you call them in any order you wish?
      7. Once you have defined an optional parameter in some specific method of your class, can you define another non-optional parameter to the right of the optional one?
      8. What should be the signature of a class method which supports receiving any number of parameters of the same type (string.Format() for example)?
      9. How can you compare that two objects are from the same type?
   6. Main namespaces
      1. Please enumerate most common used (up to 4) name spaces available in .net framework?
      2. In which namespace resides the DateTime class?
      3. In which namespace resides the Int32 class?
      4. In which namespace resides the StringBuilder class?
      5. In which namespace resides the Stream class?
      6. Is the Stream class abstract?
      7. Please tell me one class which implements the Stream class?
      8. What is the use of those classes inheriting from Stream class?
      9. In your class definition what keyword you use to import a specific namespace/library?
      10. Can I load or unload dynamically a namespace/library?
      11. What is an App Domain?
      12. In a console if you want to write a text line into output, what instruction should you use?
   7. Reflection
      1. Do you think is possible to generate code in any language or an html page from a .net assembly you have built?
      2. What is reflection in .net framework and why is so powerful?
      3. Is reflection technology available only in .net framework or is also available in other platforms like Java?
      4. What is the main method exposed by the .net framework for start reflecting an object?
      5. What are the main methods you can use when discovering/describing an object?
      6. What is code obfuscation of an assembly and how can be implemented?
      7. When should you create classes inheriting from System.Attribute class?
   8. Localization
      1. What is needed in a visual studio solution to support more than one speaking language (English and Spanish for example) when displaying text to end users?
      2. How do you handle dates so they display properly in different speaking languages (British and American English for example)?
2. C# knowledge
   1. Variable declarations
      1. What is the use of the var keyword in C#?
      2. Can you give me one reason you may declare a variable using var instead of specifying the type?
   2. Keywords
      1. Could you enumerate at least 5 C# keywords and its uses?
      2. What are the main uses of the **using** keyword/statement?
      3. Over what kind of objects works the foreach statement?
      4. How is a typical foreach statement definition?
      5. What happens when you use the break and/or continue keywords inside a for/foreach statement?
      6. Where can you initialize a private variable you’ve prefixed with the readonly keyword in your class definition?
      7. What the keyword async is used for and which other keyword needs to be set when you use it?
      8. What is the use of the static keyword in C# when applied to a class, method and field?
      9. When should you prefix a class as sealed?
   3. Static classes/methods
      1. Why you may need to use static classes in C#?
      2. What is the lifecycle of a static instance of your class?
      3. Can also a class methods or properties be prefixed as static?
      4. If you define a class method as static can you use the “this” keyword to refer to the current instance?
   4. Value and reference types
      1. What are value types and reference types?
      2. The .net framework primitive types are value types or reference types?
   5. Data types
      1. To what type maps the int type in the .net framework?
      2. To what type maps the long type in the .net framework?
      3. Please enumerate at least 5 primitive types of the .net framework.
      4. How do you handle date time types when you have several clients working on different world time zones?
      5. What date should return DateTime.MinDate?
   6. String vs StringBuilder
      1. What are immutable values in .net framework?
      2. String type is immutable or mutable?
      3. When should I use String class and when the StringBuilder class?
      4. What method of the string class you use to split a string sentence separated by spaces?
      5. What method should you use to join that split string back again?
      6. What are the most common used methods of the StringBuilder class?
   7. Collections
      1. What is the main difference between a string array and a list of strings, i.e. between arrays and collections?
      2. Please enumerate at least 4 most used .net framework collection classes.
      3. What is the difference between a list and a dictionary?
   8. Generics
      1. What are the generic classes and where does make sense to use them?
      2. Do generic class implementations improve code maintenance and readability?
      3. What is the main advantage of using generics?
      4. If you want to implement a stack class of items where they can be integers, strings or custom classes, how do you implement it using generics?
      5. What should be the header signature of a stack class of any type of items implemented using generics?
   9. LinQ
      1. What is LinQ used for?
      2. What are anonymous variables and how you define them in a LinQ select clause?
      3. When you use the let keyword in a linq expression?
      4. What namespace you need to import to use a standard array of integers with linq expressions?
      5. What linq instruction should you use to get the maximum value from an array of integers?
      6. What linq instruction should you use to get the sum of all values from an array of integers?
      7. If you have a dictionary of string keys and its values as integers, can you summarize the values grouped by keys?
      8. When you should use the First() and when you should use FirstOrDefault() methods within LinQ?
   10. Regular Expressions
       1. What is a regular expression and what is their usage?
       2. What is a regular expression pattern?
       3. How do you define a regular expression pattern to only accept alphabetic characters in lower and upper case?
       4. How do you define a regular expression pattern to only accept from 2 to 8 digits (numbers)?
   11. Exceptions
       1. What is the base class of all .net framework exceptions?
       2. How do you throw an exception in the .net framework?
       3. How do you handle exceptions in your code?
       4. What is the usage of the finally statement in an exception handling block?
       5. What common action(s) you may take when you handle an exception?
3. Algorithms
   1. Array declarations
      1. How you define an array of integers?
      2. How you make sure you initialize an array of integers before being used?
      3. If you define a List<string>, what returns the ToArray() method of that list?
      4. If you define a string[], what returns the ToList() method of that instance?
   2. Word/char counting
      1. Please describe the steps you must do to count the words on a text file?
      2. What else should I do to also fill a dictionary of letters and its incidences?
   3. String splitting
      1. What is the standard method to split a string?
      2. What do I need to do to avoid retrieving empty strings when splitting a string?
      3. When splitting a string, can I split the string using some specific string or some specific char?
   4. Word reversing
      1. Please describe the steps you must do to reverse a string?
      2. Please describe the steps you must do to sort a split string by its words length?
4. Object oriented programming
   1. Encapsulation
      1. In OOP what is encapsulation and why is one of the pillars of OOP?
      2. In .net framework what is the visibility of a private var?
      3. In .net framework what is the usage of a protected var?
      4. What is shadowing of a class property in .net framework?
      5. In .net framework, what is the standard definition of a public property getter, setter (int property for example)?
      6. What is a constructor and a destructor of a class?
      7. Can constructors be declared as private in C#?
      8. Can multiple constructors be declared in C#?
   2. Inheritance
      1. What is the meaning of the inheritance term in OOP?
      2. What is a base class?
      3. What is a root class of a hierarchy of classes?
      4. Why you may need to implement a hierarchy of classes?
      5. What (items) should be defined in a base class?
      6. What (methods) should be implemented in a base class?
      7. In .net framework what C# keyword you use to define your class can not be instantiated?
      8. In .net framework what C# keyword you use to define your class can not be inherited anymore?
      9. In .net framework what C# keyword you use to define your derived classes must implement a specific method?
   3. Polymorphism
      1. What is the meaning of the polymorphism term in OOO?
      2. How the polymorphism is implemented in a derived class?
      3. How do you define optional implementation and how you define strict implementation of a derived class method?
   4. Loosely/Tightly coupled
      1. What is a loosely coupled design of classes and what are its main benefits?
   5. Interfaces
      1. What is an interface in OOO?
      2. What C# keyword is used to define an interface?
      3. What items can be defined in an interface?
      4. In .net framework, can properties be declared in an interface?
      5. Can interface methods be declared as private or protected?
      6. When you use an interface instead of an abstract class?
      7. What are the main differences between an interface and an abstract class?
   6. SOLID
      1. In software development SOLID what the S stands for
      2. In software development SOLID what the O stands for
      3. In software development SOLID what the L stands for
      4. In software development SOLID what the I stands for
      5. In software development SOLID what the D stands for
5. Patterns
   1. Concepts
      1. What is the meaning of software patterns, what its benefits?
   2. Factory method
      1. Please describe in what cases is recommended to implement the factory method pattern
      2. Please enumerate at least 2 classes in .net framework or third party components which implement the factory method pattern
   3. Singleton
      1. Please describe what is the singleton pattern and when should be used?
      2. Please describe the standard implementation of the singleton pattern
   4. Builder
      1. Please describe in what cases is recommended to implement the builder pattern
      2. Please enumerate at least 2 classes/frameworks/platforms you know implement the builder pattern
6. Application Design
   1. Performance/Caching
      1. What should be an acceptable time for a web page to be rendered in a web site?
      2. What is the caching technology and how it may be implemented?
      3. What items should be cached in an application?
      4. What may be the difference between a Session/Application (web development) and a cache store?
      5. Please enumerate at least 2 ways in .net framework to implement caching.
   2. Messaging
      1. What is the main usage of the messaging technology?
      2. In what cases you should incorporate messaging within your application?
      3. Please enumerate at least 2 technologies/platforms for messaging.
   3. Scheduling
      1. Why scheduling technology should be used in complex systems/services?
      2. Please enumerate at least 2 technologies/platforms for scheduling.
      3. What is a standard way to implement in .net framework a scheduled service, what kind of visual studio project can be used?
7. Web Development
   1. Web Concepts
      1. What is a Request?
      2. What is a Response?
      3. What are the Header attributes and what is its use?
      4. What is the lifecycle of a Request?
      5. Why we said the web is stateless?
      6. Please enumerate the principal http verbs when making a request to a server and its use.
      7. Please enumerate the most common Response statuses
   2. Web API
      1. What is an API controller?
      2. What is the base class of the API controller?
      3. What is a Web API Route?
      4. How do you define a Route for a controller method?
      5. What is the use of the Response Type attribute for a controller method?
   3. MVC Development
   4. WCF
8. SQL Development
   1. Concepts
      1. What is a View?
      2. What is a Primary Key?
      3. What is an Index?
      4. What is the difference between a Primary Key and an Index?
      5. What is the difference between a SQL function and a store procedure?
      6. What is the Normalization Term in SQL?
      7. Should you always have normalized schemas or may not apply in all circumstances?
      8. What is a Trigger and when is executed?
      9. What is a SQL transacted operation and its common statements?
      10. When you should run SQL operations using transaction statements?
      11. What are the common CRUD SQL operation statements?
9. Unit Testing
   1. Concepts
   2. Frameworks