Interview Questions
That some CS students should be able to answer to

O. Ozbek

2021

Introduction

This piece of document is a collection of questions that I was asked to answer, or I have asked professionally during recruitment. The idea behind this is to look more into the contents if you are not comfortable enough with the small description. For example, if you don't know what something does, you should look it up and try to implement it in a project. I'm mostly adding questions related to my field and my interview experiences.

Contents

Questions

0.1   Programming Related

These are usually C#, Python and OOP related questions. Since most terms are similar, like Lambda and Func, it should be generic enough for most to understand, considering most of us either take C#, C++, Java or Python classes.

What does the Static keyword do?
Static members belong to the class instead of a specific instance.

What is a subclass?
A subclass is a class that describes the members of a particular subset of the original set. They share many of characteristics of the main class, but may have properties or methods that are unique to members of the subclass. Cat could be a subclass for Animal class.

Why do we use the Virtual keyword?
The virtual keyword is used to modify a method, property, indexer, or event declaration and allow for it to be overridden in a derived class. This method can be overridden by any class that inherits it.

What is an Interface?
An interface contains definitions for a group of related functionalities that a non-abstract class or a struct must implement. An interface may define static methods, which must have an implementation. An interface may not declare instance data such as fields, auto-implemented properties, or property-like events. By using interfaces, you can include behavior from multiple sources in a class. That capability is important because some languages don't support multiple inheritance of classes. In addition, you must use an interface if you want to simulate inheritance for structs, because they can't actually inherit from another struct or class.

What is an Event?

Events enable a class or object to notify other classes or objects when something of interest occurs. The class that raises the event is called the publisher and the classes that handle the event are called subscribers.

What happens if an Event is called but nothing subscribed to it yet?

Rest of the methods will not execute, the exception propagates to a higher level. Unless it is handled at the top level, the application might crash.

What is the difference between an Action and a Func?

Action is a standard delegate that has one to four parameters and doesn't return value. It's used to represent an action. Func is the more generic one, it has the same amount parameters and returns something. Both their parameters are 16 in .NET 4.

What is a Lambda function?

It is a simple, short throwaway function which is designed to be created in-line. They are also known as anonymous functions. It is intended for a single use.

What is the difference between a Struct and a Class?

Structs are value types, allocated either on the stack or inline in containing types while Classes are reference types, allocated on the heap and garbage-collected.

Allocations and de-allocations of value types are in general cheaper than allocations and de-allocations of reference types. Assignments of large reference types are cheaper than assignments of large value types. In structs, each variable contains its own copy of the data *(except in the case of the ref and out parameter variables)*, and an operation on one variable does not affect another variable. In classes, two variables can contain the reference of the same object and any operation on one variable can affect another variable. In this way, struct should be used only when you are sure that, it logically represents a single value

like primitive types, it is immutable, and it should not be boxed and un-boxed frequently. In all other cases, you should define your types as classes.

What is an Optional?

An optional is a kind of container. An optional String is a container which might contain a String. An optional Int is a container which might contain an Int. Think of an optional as a kind of parcel. Before you open it, or "unwrap", you won't know if it contains something or nothing.

What is the difference between an Array and a List?

A List can consist of elements belonging to different data types while an array only consists of elements belonging to the same data type. Arrays can directly handle arithmetic operations. Lists can be nested to contain different type of elements while Arrays must contain either all nested elements of same size. List also has a greater flexibility that allows easy modification of data. An array is comparatively more compact in memory size. In some languages, Array size is defined.

What is a Singleton?

A singleton is simply a class which only allows a single instance. You can't new up instances of a singleton class. Static classes in C# are a good example of singleton behavior because, in most cases, there can only ever be one instance of a static class. You should definitely check how to implement a singleton class without the use of a keyword.

How do you remove things from a List that is being iterated?

You can use a list comprehension to create a new list containing only the elements you don't want to remove. Or, by assigning to the slice, you can mutate the existing list to contain only the items you want. This approach could be useful if there are other references to that list that need to reflect the changes. Instead of a comprehension, you could also use itertools in Python

What is abstraction?

Abstraction is to handle complexity by hiding unnecessary details from the user. That enables the user to implement more complex logic on top of the provided abstraction without understanding or even thinking about all the hidden complexity.

What is polymorphism?

To implement multiple methods within the same class that use the same name but a different set of parameters. That is called method overloading and represents a static form of polymorphism. Within an inheritance hierarchy, a subclass can override a method of its superclass. That enables the developer of the subclass to customize or completely replace the behavior of that method.

What is the difference between a Stack and a Queue?

A stack is a data structure that contains many elements. Stacks are linear, that is, there are a sequence and an order to how they are constructed and traversed. Stacks only have one direction, if we wanted to add elements to this structure, or if we wanted to remove items from it, we would have to do it starting from one place: the top of the stack.

A queue is a linear abstract data type that can contain a long list of elements. But what's important to remember about them is how they can grow and shrink in size. Queues abide by the First In, First Out (FIFO) principle.

What is hashing?

Hashing is designed to solve the problem of needing to efficiently find or store an item in a collection. Hashing means using some function or algorithm to map object data to some representative integer value. This so-called hash code can then be used as a way to narrow down our search when looking for the item in the map. Generally, these hash codes are used to generate an index, at which the value is stored.

What is the time complexity of Linear Search?

O(n), since it checks each element of the list sequentially until a match is found.

How many bits is an Int?

32 bits, but it depends on compilers from time to time.

What is ADO.NET?

ADO.NET provides consistent access to data sources such as SQL Server and XML, and to data sources exposed through OLE DB and ODBC. Data-sharing consumer applications can use ADO.NET to connect to these data sources and retrieve, handle, and update the data that they contain. ADO.NET separates data access from data manipulation into discrete components that can be used separately or in tandem.

0.2   Version Control

These are mostly about Git and GitHub. Many other applications such as Heroku or SourceTree use Git or a similar CLI anyway, so these should be good to know.

What is a Repository?

Repository is a central file storage location. It is used by version control systems to store multiple versions of files. While a repository can be configured on a local machine for a single user, it is often stored on a server, which can be accessed by multiple users.

What is Commit, Push & Pull?

Commit creates a new snapshot of the project, pointing to updated directories, files, etc. and one or more parent commits.

Push sends a set of commits and the objects they point to to a remote.

Pull copies a set of commits and referred objects from a remote."

What is a merge conflict?

Merge conflicts happen when you merge branches that have competing commits, such as changes in same file or line.

What are local, remote, and tracking branches?

A local branch is a branch that only the local user can see. It exists only on your local machine.

A remote branch is a branch on a remote location, usually pushed to origin.

A remote tracking branch is a local copy of a remote branch.

A local tracking branch is a local branch that is tracking another branch.

What is cherry-pick commit?

Cherry picking in Git means to choose a commit from one branch and apply it onto another.

What is Origin?

"Origin" is the name of the remote repository where you want to publish you commits. By convention, the default remote repository is called "Origin", but you can work with several remotes at the same time.

0.3   Debugging

What is a Breakpoint?

Breakpoints are one of the most important debugging techniques in your developer's toolbox. You set breakpoints wherever you want to pause debugger execution. For example, you may want to see the state of code variables or look at the call stack at a certain breakpoint.

What are the different stepping options?

Step Into is for when a method that is about to be invoked, and you want to debug into the code of that method, so the next step is to go into that method and continue debugging step-by-step.

Step Over is for when a method is about to be invoked, but you're not interested in debugging this particular invocation, so you want the debugger to execute that method completely as one entire step.

Step Return is for when you're done debugging this method step-by-step, and you just want the debugger to run the entire method until it returns as one entire step.

0.4 Unity Related

What is Unity's life cycle?
Awake, Start, Update, LateUpdate, OnDestroy. There are also event calls and parallel updates like FixedUpdate.

What is the difference between Update and FixedUpdate?
Update runs once per frame. FixedUpdate can run once, zero, or several times per frame, depending on how many physics frames per second are set in the time settings, and how fast/slow the framerate is.

What is DontDestroyOnLoad?
The load of a new Scene destroys all current Scene objects. Call Object.DontDestroyOnLoad to preserve an Object during level loading. If the target Object is a component or GameObject, Unity also preserves all of the Transform's children. Object.DontDestroyOnLoad does not return a value.

When do we use OnValidate?
You can use this to ensure that when you modify data in an editor, that data stays within a certain range.

How do we write custom inspectors for MonoBehaviour?
Create another class inside the existing class so it could access the private members like "public class StackPreview : Editor". You'll see the contents when you run.

What are Quaternion Angles?

Quaternions provide a convenient mathematical notation for representing spatial orientations and rotations of elements in three dimensional space. Think of two GameObjects (A and B) moving around a third GameObject (C). Lines from C to A and C to B create a triangle which can change over time. The angle between CA and CB is the value Quaternion.Angle provides.

You can also think of them as 3D objects rotated states in 4D, that is why Quaternions have 4 parameters.

What are RectTransform, AnchorPoint and Pivot?

RectTransform mainly provides the position, size, anchor point and center information of a rectangle, as well as methods to manipulate these properties. Anchor is useful for ensuring that your UI will maintain its position and size relative to the screen size. Pivot is the normalized position in this RectTransform that it rotates around.

What is a Shader?

A shader is a piece of code, that is executed on the GPU. The engine feeds it with 3d model vertices, textures and other information, and gets back from it pixel colours.

What is a Material?

A material is a separate asset in your project which contains a collection of settings.

What is a Fragment Shader?

A Fragment Shader is the Shader stage that will process a Fragment generated by the Rasterization into a set of colors and a single depth value. The output of a fragment shader is a depth value, a possible stencil value (unmodified by the fragment shader), and zero or more color values to be potentially written to the buffers in the current framebuffers. Fragment shaders take a single fragment as input and produce a single fragment as output.

What is a Vertex Shader?

The Vertex Shader is the programmable Shader stage in the rendering pipeline that handles the processing of individual vertices. Vertex shaders are fed Vertex Attribute data, as specified from a vertex array object by a drawing command. A vertex shader receives a single vertex from the vertex stream and generates a single vertex to the output vertex stream. There must be a 1:1 mapping from input vertices to output vertices. Vertex shaders typically perform transformations to post-projection space, for consumption by the Vertex Post-Processing stage. They can also be used to do per-vertex lighting, or to perform setup work for later shader stages.

0.5 Data Science

What is the difference between data mining and data exploration?
Data exploration is a way of using the results gathered by data mining. The end result is different.

Can you describe the different types of data analysis?
There are 4 that I know of, that are mostly used. Descriptive (What happened), Diagnostic(Why), Predictive(What could happen) and, Prescriptive(What is the best approach).

What steps does Data Analysis normally consist of?
Define, Collect, Clean, Analyze, Apply. You define why you need the data analysis, collect data from sources, clean corrupted or unnecessary data, analyze the data and, produce results.

Can you explain what normal distribution is?
Normal distribution is easier to explain with an example. Let's say all employees from a pencil company sold 75 pencils on average. For normal distribution to be applicable, half of our employees sold less than 75 while the rest sold more than 75, but most of them sold about 75.

What are the different types of hypothesis testing?

Parametric and non-parametric test. A parametric test is used when we make a specific assumption about the underlying distribution of the population from which the sample is being drawn, and which is being investigated.

A non-parametric test is a hypothesis test that does not require the population's distribution to be characterized by certain parameters.

0.6 Web Applications

Which layer does the HTTP work on?

The Hypertext Transfer Protocol is an application layer protocol

When would you use SVG instead of a Canvas?

If you want to support more resolutions and be responsive. The size can expand beyond its natural.

If you need event handlers, since Canvas does not provide the support for event handlers.

SVG refers to draw, Canvas refers to paint. Therefore SVG would not be the better option for gaming applications.

What does display: inline and display: block do?

Compared to display: inline, the major difference is that display: inline-block allows to set a width and height on the element.

Also, with display: inline-block, the top and bottom margins & paddings are respected, but with display: inline they are not.

Compared to display: block, the major difference is that display: inline-block does not add a line-break after the element, so the element can sit next to other elements.

What are position relative, absolute and fixed?

If you set position: relative; on an element but no other positioning attributes (top, left, bottom or right), it will have

no effect on it's positioning at all, it will be exactly as it would be if you left it as position: static;

You use the positioning attributes top, left, bottom, and right to set the location for position: absolute. These values will be relative to the next parent element with relative (or absolute) positioning. If there is no such parent, it will default all the way back up to the ¡html¿ element itself.

A fixed position element is positioned relative to the viewport, or the browser window itself. The viewport doesn't change when the window is scrolled, so a fixed positioned element will stay right where it is when the page is scrolled.

What is lazy loading?

Lazy loading is a strategy to identify resources as non-blocking (non-critical) and load these only when needed. It's a way to shorten the length of the critical rendering path, which translates into reduced page load times.

What are GET, POST, PUT, PATCH and DELETE?

GET, POST, PUT, PATCH, and DELETE are the five most common HTTP methods for retrieving from and sending data to a server.

The GET method is used to retrieve data from the server. This is a read-only method, so it has no risk of mutating or corrupting the data.

The POST method sends data to the server and creates a new resource. The resource it creates is subordinate to some other parent resource.

The PUT method is most often used to update an existing resource.

The PATCH method is very similar to the PUT method because it also modifies an existing resource. The difference is that for the PUT method, the request body contains the complete new version, whereas for the PATCH method, the request body only needs to contain the specific changes to the resource, specifically a set of instructions describing how that resource should

be changed.

The DELETE method is used to delete a resource specified by its URI.

What are cookies?

The purpose of the computer cookie is to help the website keep track of your visits and activity. A computer cookie consists of information. When you visit a website, the website sends the cookie to your computer. Your computer stores it in a file located inside your web browser.

Different types of cookies keep track of different activities. Session cookies are used only when a person is actively navigating a website; once you leave the site, the session cookie disappears. Tracking cookies may be used to create long-term records of multiple visits to the same site. Authentication cookies track whether a user is logged in, and if so, under what name.

What is TOTP and HOTP?

HOTP: Event-based One-Time Password, TOTP: Time-based One-Time Password.

What are closures in JS?

A closure gives you access to an outer function's scope from an inner function. In JavaScript, closures are created every time a function is created, at function creation time.

0.7 Database

What is index in SQL?

An index is a schema object. It is used by the server to speed up the retrieval of rows by using a pointer. It can reduce disk I/O(input/output) by using a rapid path access method to locate data quickly. An index helps to speed up select queries and where clauses, but it slows down data input, with the update and the insert statements. Indexes can be created or dropped with no effect on the data.

What is cross-join?

It produces a result set which is the number of rows in the first table multiplied by the number of rows in the second table if no WHERE clause is used along with CROSS JOIN. This kind of result is called as Cartesian Product. If WHERE clause is used with CROSS JOIN, it functions like an INNER JOIN.

What is a transaction?

A transaction is a single unit of work. If a transaction is successful, all of the data modifications made during the transaction are committed and become a permanent part of the database. If a transaction encounters errors and must be canceled or rolled back, then all of the data modifications are erased.

What are isolation levels?

Read Uncommitted, Read Committed, Repeatable Read, Snapshot, Serializable

0.8   Architecture

What are SOLID principles?

Single-Responsiblity, Open-Closed, Liskov Substitution, Interface Segregation, Dependency Inversion.

What are the design principles for OOP?

Abstraction, Encapsulation, Hierarchy (Association, Aggregation), Inheritance, Polymorphism. *You should know these.*

0.9   Project Managing

What is a Gantt Chart?

A Gantt chart is a type of bar chart that illustrates a project schedule, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale.

Usually, a vertical line would signal today and a diamond would signal a milestone.

What is the main loop of a Project Manager?

The planning, resourcing, managing and, motivating. Managing part can be divided to *monitor, reporting, controlling and, requesting.* This should respect the software development life cycle.

What are resources?

Materials, People and Cost could be a good summary. Employees, Hardware, Facilities, Software, Budgets, Time are all good examples for resources.

What will you gain with a good resource management or a schedule?

When a reallocated, cross-referenced working hour schedule is built, it will decrease the chances for a problem in case a resource ends up being unavailable. It allows for buffer-time, prevents burnout, is weighed evenly and, empty spaces are filled.

What makes a good Post-Mortem Meeting?

These meetings are often awkward and unproductive because everyone leaves knowing what they did wrong, but not what to do next. Post-mortems create a feedback loop that helps your agency to retain current and future clients. Productive post-mortems help illuminate the little things, like the fact that your clients want to see an earlier draft of the wireframe or require project updates more frequently. Post-Mortems should be consistent and productive.

Good scheduling, moderation and inclusion of clients usually make a good Post-Mortem. Encouraging honest critical discussion, asking "why?", keeping the feedback constructive and recording actionable takeaways are key. Follow-up mails for clients regarding their feedback also ensures a healthy relationship.

What is a project charter?

A project charter is a short document that sets out the core details of the project's purpose and aims. It's created in line with the business case for the project as described by the project sponsor (i.e. the person who's responsible for the project's existence in the first place), and it's a crucial part of the initiation process at the start of the project. Ultimately, your project charter documents the goals, objectives, and business case for the work. Once created, the project charter serves as a sort of contract between the project sponsor and the project manager, giving a top-level overview of the expectations and authorizing the project manager to mobilize resources to achieve the right outcomes.

What are some key management areas for Project Managers?

Integration management, Scope management, Schedule management, Cost management, Quality management, Resource management, Risk management, Communications management, Procurement management, Stakeholder management

0.10   Common Tools & Tech

What is Docker?

Docker is an open-source project for automating the deployment of applications as portable, self-sufficient containers that can run on the cloud or on-premises.

What is Kubernetes?

Kubernetes is an open-source container-orchestration system for automating computer application deployment, scaling, and management. It was originally designed by Google.

What is MATLAB?

MATLAB is a proprietary multi-paradigm programming language and numeric computing environment developed by MathWorks. MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user

interfaces, and interfacing with programs written in other languages.

What is AWS?

Amazon Web Services is a cloud service from Amazon, which provides services in the form of building blocks, these building blocks can be used to create and deploy any type of application in the cloud. These services or building blocks are designed to work with each other, and result in applications that are sophisticated and highly scalable.