Technical Assessment

Junior DevOps

Sava Tudor 26.03.2024

1. Exemplify two data structures that you know and describe some situations where you would use them.

Answer: I think that the most common data structure used in programming is a one-dimensional Array. It stores a collection of elements that can be duplicated in a continuous memory space and allows for the retrieval of an elements by it’s index in O(1) complexity.

Another data structure that is very similar to an array is the Linked-List. It also stores a collection of elements that can be duplicated, but not in a continuous memory space. Instead, it uses pointers so that each element of the list points to the next one, and thus it does not need a single united memory space. And as such, it is more memory efficient because it only stores the elements that it has and adding or removing elements is less complex than it is in an array.

So, if I were in a situation where I would do a lot of accessing of an element I would use an array, and I were in a situation where I would do more adding/deleting from a collection I would use a linked list.

1. You open a web browser and access http://www.tss-yonder.com. What is the IP address behind this website and how does the browser know how to get the correct IP?

Answer: A quick way of finding the IP address behind a website is to ping it in the command prompt.

A computer screen with white text

Description automatically generated

Thus, we can see the IPv4 address of a website. But it’s difficult for a person to remember such and address and a new way of linking actual names to IP addresses has been implemented. It uses a DNS (Domain Name System) Server which maps the IP address to the name of a website. So, when a user tries to access in a browser <http://www.tss-yonder.com>, first the browser will make a request to a DNS server to find out the actual IP address, and then it will load the page that is present at that address.

1. Exemplify two transport protocols and think of two applications that would use each of them.

Answer: TCP (Transmission control protocol) is the most used transport protocol, and it requires an established connection before sending any data. It divides the data into segments that are then assigned a header and meta-data and sent through the network and has some implemented methods to ensure the secure and integral transfer of data. HTTP, FTP, and mail service are some of the applications that are based on TCP.

The other major transport protocol is UDP (User datagram protocol) which does not require a connection established between the sender and the receiver and there is no guarantee of the integrity of the data that is being sent. It is faster than TCP and thus it is used for media streaming, online games and other applications that require fast transfer of data.

1. You wrote a chat web application in your favorite programming language. You need to host this somewhere and run it so that the entire world can start using it. Describe how you would do that and the tools you would use.

Answer: Being a small, personal project, I would opt to have it deployed on some cloud provider like AWS or Azure. The lightest and straightforward option for me would be to create a Docker image that would then be deployed on the cloud, so that each client can send messages. Another part that needs to be configured is the DNS and the actual name of the website with which the users can access it, for which I would probably use one of the many online platforms available for this purpose.

1. Now your application is famous but unfortunately it has a lot of bugs. You want only you and a couple of your friends to be able to access it until you patch it. Describe two ways you can achieve this.

Answer: One option would be to introduce a login system where only the person working on the application would have access to. Another option would be to remove the application from the internet and work on it locally, using version controls systems such as Git with GitHub to combine all the changes of the persons working on the application and when the application reaches a satisfactory point for the “stakeholders”.

1. Your application is ready for the public once again. You realize that you forgot about security and any network administrator can see the messages that a user sends or receives. How would you improve your application to prevent this? Is there any way to do this so that not even the application owner (you) can see the messages between two random users?

Answer: Encryption would be the answer to this question. I would implement encryption algorithms so that only the sender and the receiver can see the unencrypted messages. This implies that all the data that is being sent to the network should be encrypted end-to-end and the keys to encrypt/decrypt such messages would depend on the user’s account/ hardware that he uses.

1. What are cookies and what are they used for? Find a cookie used by http://www.tss-yonder.com and copy its name and value. What do you think is its purpose?

Answer: Cookies are extra data sent to a web client by the server regarding authentication or personal preferences that the user has selected before.

Cookie found in the Yonder website:

A close up of a text

Description automatically generated

Cookies are used to facilitate the experience of the users by remembering that he has logged in previously and skipping that step when revisiting the page, remembering certain settings that he changed. Also, they can be used to gather data on what the user has clicked/ viewed and transmit that information to interested 3rd parties.

1. While writing your application you need to create more worker processes for processing some data. How can you create child processes in your favorite language? What are the possible states of a process?

Answer: In Java, a neat and easy way to create more worker processes for processing some data is to use Executor Services. You just give it the number of threads you want it to use, the action that you want them to execute and just execute the entire service. Possible states of a process include running, waiting (maybe some resource that it needs is blocked by another thread), ended, hanging (when for example a deadlock is created, and two processes wait on each other, and they never end).

1. Your application is running but it still has a few problems. Occasionally, it returns an error page. How can you find the PID of your application? What would you do to debug it?

Answer: Logging could be an answer. If we log the calls that are being made and we identify which one returns an error and show its stack trace, we could debug and fix the problem. The PID could be retrieved from using Java commands.

1. What DBMS would you use to store your application data and why? How would you store the passwords of each user?

Answer: Depending on the scale and complexity which I would want the application to become, I would have to compare Sql to NoSql options and see which one is best suited for my application. But the passwords of each user I would hash, so that if a data breach happens, the attacker cannot see the passwords and login impersonating other users.