Ternary search:

Instead of dividing the array to two parts we divide it to 3 parts

L=l+(r-l)/3

R=r-(r-l)/3

Which gives us complexity log n base 3 instead of 2

Code c ++:

```
using namespace std;
int ternarySearch(int l, int r, int key, int ar[])
{
    if (r >= l) {

        // Find the mid1 and mid2
        int mid1 = l + (r - l) / 3;
        int mid2 = r - (r - l) / 3;

        // Check if key is present at any mid
        if (ar[mid1] == key) {
            return mid1;
        }
        if (ar[mid2] == key) {
            return mid2;
        }


        if (key < ar[mid1]) {

            // The key lies in between l and mid1
            return ternarySearch(l, mid1 - 1, key, ar);
        }
        else if (key > ar[mid2]) {

            // The key lies in between mid2 and r
            return ternarySearch(mid2 + 1, r, key, ar);
        }
        else {

            return ternarySearch(mid1 + 1, mid2 - 1, key, ar);
```

Types of design patterns

1. Creational: These patterns are designed for class instantiation. They can be either class-creation patterns or object-creational patterns.

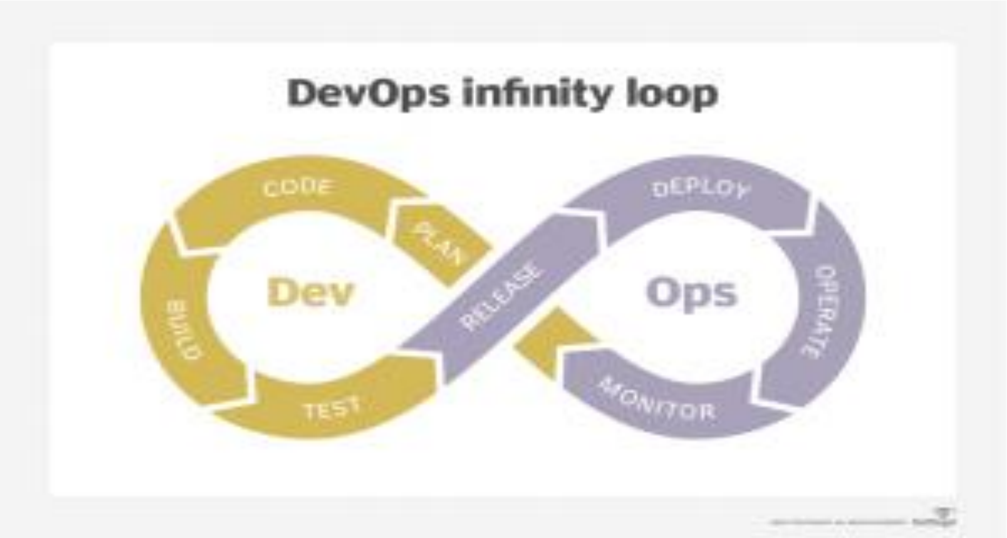
2. Structural: These patterns are designed with regard to a class's structure and composition. The main goal of most of these patterns is to increase the functionality of the class(es) involved, without changing much of its composition.

3. Behavioral: These patterns are designed depending on how one class communicates with others.

Agile vs Devops

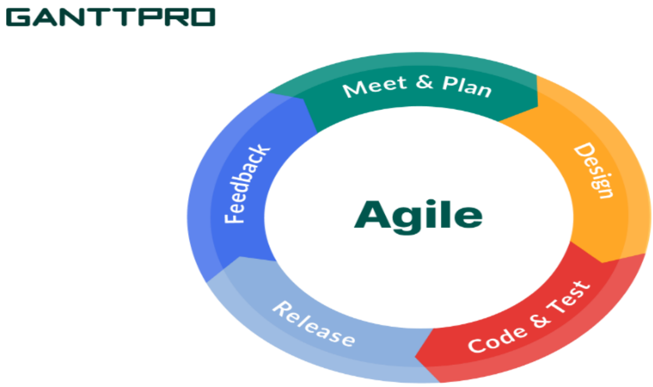
Devops:

it’s ***continuous deployment***.



Agile:

highest priority is ***continuous delivery***



What are used tools in DevOps

Plan:

Slack-jira software-confluence

Build:

Git-jira-maven

Testing:

Selenium

Release:

Docker-saltstack

Congiguiring and operating:

Chef-kubernetes-puppet-ansible

Monitoring:

Splunk-nagios