

Tổng hợp Kiến Thức C# / .NET Nâng Cao

1. Memory & Performance

- Stack vs Heap (đặc điểm, khi nào lưu ở đâu)
- Value Type vs Reference Type
- Boxing / Unboxing (hiệu suất, cách tránh)
- List<T> vs ArrayList (tối ưu generic)
- Span<T>, ref struct (tránh allocate, dùng stack)
- Object Pooling (reuse object tránh GC)
- Manual GC Handling: ``GC.Collect``, ``SuppressFinalize``, ``KeepAlive``
- LOH (Large Object Heap), tránh cấp phát lớn

2. Multithreading / Concurrency

- Thread vs ThreadPool
- Task / Task<T>
- async / await (không tạo thread mới, context switching)
- TaskCompletionSource, WhenAny, Delay
- Parallel.ForEach, Parallel LINQ (PLINQ)
- SemaphoreSlim, lock, Monitor, Interlocked
- Dataflow blocks, Channel<T>

3. Redis / Memcached / Cache

- Redis: sorted set, pub/sub, stream, ranking, TTL, eviction, clustering
- Redis persist: RDB vs AOF
- Redis security: Auth, TLS, ACL
- Redis message queue: dùng list/stream
- Memcached: key-value thuần volatile, scale thủ công, TTL

4. Struct vs Class

- Value type vs Reference type
- Immutable / Mutable struct
- Copy vs reference behavior

- Truyền ref/out/in vào method
- Dễ gây boxing khi dùng interface/object

5. GC nâng cao

- Gen 0 / 1 / 2
- Khi nào GC chạy
- Finalizer (~ClassName), delay thu gom
- IDisposable, `using`, giải phóng resource unmanaged đúng lúc
- GC Safe Points, Full GC Notification

6. Thread-Safe Patterns & Synchronization

- Thread-safe collection
- lock, ReaderWriterLockSlim
- volatile keyword
- double-check locking pattern
- ConcurrentQueue, Dictionary, Stack

7. Dependency Injection & Middleware

- Service lifetime: Singleton, Scoped, Transient
- Custom middleware
- Filter, ModelBinder tùy chỉnh
- IHostedService, BackgroundService
- Service Provider / Factory pattern

8. Reflection / Dynamic / Emit

- Reflection, Type.GetType(), MethodInfo
- Expression<T>, compile expression
- Reflection.Emit / IL generation
- Source Generator (compile-time code gen)
- DynamicMethod / IL rewiring

9. System Design nâng cao

- MessageQueue: RabbitMQ, Redis Stream

- Event-Driven Architecture
- CQRS, Event Sourcing
- API Gateway, Retry/Timeout/Backoff
- Circuit Breaker, Resilient Policy (Polly)

10. Advanced Hosting & Deployment

- NativeAOT (build native exe, no runtime)
- Trimmed publish, ReadyToRun, single-file
- Docker, K8s, systemd, CI/CD
- Health check, metrics, OpenTelemetry

11. Bảo mật

- Auth: JWT, OAuth2, OpenID Connect
- Auth handler, policy-based auth
- TLS, DataProtection API
- Secure token rotation
- Input validation, anti-XSS/CSRF

12. Observability & Debug

- Logging: Serilog, structured logs
- Tracing: OpenTelemetry, DiagnosticSource
- dotnet-trace, PerfView, BenchmarkDotNet
- Monitor lệnh Redis (MONITOR)

13. Plugin / Modular Architecture

- AssemblyLoadContext (hot reload plugin)
- Domain isolation
- Viết SDK / công cụ dạng plugin
- Interface-based plugin + reflection

14. High-performance App

- System.IO.Pipelines
- SocketAsyncEventArgs

- BufferWriter<T>, Utf8Json, MessagePack
- Avoid closure / allocate trong delegate
- LoggerMessage.Define()

15. Cross-platform / WASM / IoT

- Blazor WASM / WebAssembly for .NET
- .NET IoT (GPIO, Serial, Sensor)
- Build lib native cross OS
- Write to target Linux, Windows, WASI

16. Tooling & Meta Programming

- Roslyn Analyzer / Code Fix / Diagnostic
- Viết DSL riêng bằng C#
- Source Generator
- Tự kiểm tra quy tắc compile time