

WIE NO WOLFSO LEVELS: NO SECRET EXTE if ((gamemode == commercial)

= R_FlatNumForName (SKYFLATNAME)

remode != commercial) ! (skill == sk_nightmare II respawiparm) respaymenters = true (fastparm II (skill == sk. nightmare && gameskill != sk. nightmare) for (i=S SARG RUN1 i<=\$ SARG PAIN2); i++1 mobjetoIMT_BRUISERSHOTI.speed = 20°FRACUNIT mobjefolMT_HEADSHOTI.speed = 20*FRACUNIT; mobjefelMT_TROOPSHOTLspeed = 20°FRACUNIT for (i=S SARG RUN1 i <= \$ SARG PAIN2; i++) stateshillins < <= 1 mobjetoIMT_BRUISERSHOTE.speed = 15 PRACUNIT mobjinfolMT_HEADSHOTLspeed = 10 FRACUNIT; mobjinfolMT_TROOPSHOTLspeed = 10 FRACUNIT // force players to be initialized upon first level load for (i=0; i<MAXPLAYERS i++1 playershil.playerstate = PST REBORN

demoplayback = false viewactive = true,

gameepisode = episode

viewactive = true

S. ResumeSound ():

e messy with SPECIAL and commented parts.

if (skill > sk_nightmare)

acks to make the latest edition work. // will be set talse if a demo

(SPR SHTG.1.5 INULLES SGUNDED II). HT2.1.7,(NULL) S DSNR2D,(II)

(SPR_PISE32768,7,(A_Light1),S_LIGHTDONE.D,O),

SPR SHTG.0.3 INULLES SCUNZ (L.O.). // S SCUNT

SPR_SHTG.0.1 (A Lower),S SGUNDOWN,D,D), // S SGUNDOWN

// do things to

switch (

break

G Do

case d

case d

Agenda

Multithreading
Concurrency
Threads
Task Parallel Library
Asynchronous Programming

Multithreading

Enables executing several pieces of code simultaneously

Leverage multicore CPUs

Speed

The operating system decides the order

Concurrency

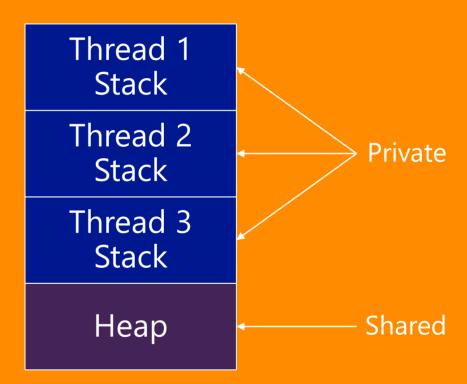
A property of systems in which several computations are executing simultaneously, and potentially interacting with each other. The computations may be executing on multiple cores in the same chip, preemptively time-shared threads on the same processor, or executed on physically separated processors.

Threads

Stack

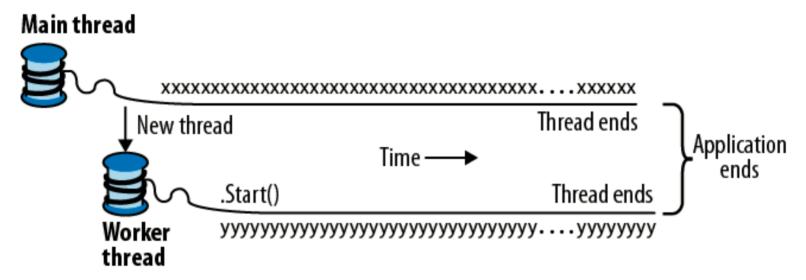
Heap

Single Threaded Program



Multithreaded Program

Threads Example



© From C# 5.0 in a NUTSHELL

Threads Demo



Race Condition

Behavior of a program where the output is dependent on the sequence or timing of other uncontrollable events.

→ Bug, when events do not happen in the order the programmer intended.

Race Condition Demo



Deadlock

A situation in which two or more competing actions are each waiting for the other to finish, and thus neither ever does.

Deadlock demo

Task Parallel Library

Task.Run Task.Factory... Task.Delay

Parallel.For

Parallel.ForEach

Parallel.Invoke

Parallel Linq → .AsParallel()

Task Parallel Library demo

System.Collections.Concurrent

ConcurrentQueue<T>

ConcurrentStack<T>

BlockingCollection<T>

ConcurrentDictionary<TKey, TValue>

Asynchronous Programming

async →

Method must return **void**, **Task**, **Task<T>**, or a task-like type.

Specifically: a type, which satisfy the **async** pattern, meaning a **GetAwaiter** method must be accessible.

await → Await task(s)...

Note: Test methods must return Task

Async demo