Bijlage Ontwerpen (Design Patterns):

Je mag volgende **afkortingen** voor het klassendiagram, het schrijven van de implementatie (= Java code) van de design patterns gebruiken:

A	abstract	b	boolean
I	interface	c	char
		d	double
v	void	i	int
		S	String

Volgende **afkortingen** voor het schrijven van de testen en de implementatie (= Java code) van de design patterns:

C	class	pu	public
		pro	protected
		pri	private
r	return	t	true
n	new	f	false

HO GENT

Bijlage Java:

Concurrency

Interface Runnable
public void run()

Interface Callable<V>
public V call()

Interface Future<V>
public V get()

Class Thread
public Thread(Runnable target)
public static Thread currentThread()
public void interrupt()
public final boolean isAlive()
public final void join()
public final void join(long millis)
public static void sleep(long millis)
public void start()

Class Executors
public static ExecutorService newFixedThreadPool(int nThreads)

Interface ExecutorService extends Executor
public void execute(Runnable command**)**
public void shutdown()
public <T> Future<T> submit(Callable<T> task)

Interface Lock
public void lock()
public Condition newCondition()
public void unlock()

Class ReentrantLock
public ReentrantLock()

Interface Condition
public void await()
public void signal()

Class ArrayBlockingQueue<E>
ArrayBlockingQueue(int capacity)
public E poll(long timeout, TimeUnit unit)
public void put(E e)
public E take()

Enum Class TimeUnit
MILLISECONDS

Class Platform
public static void runLater(Runnable runnable)

HO GENT

HO
GENT