Homework Appello di Settembre

Supponete di essere in ambiente Java Micro Edition, precisamente CLDC1.1 (https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.1/jsr139/index.html)

Supponete di voler utilizzare in questo ambiente una libreria di classi (myLib) nata in ambiente J2SE 1.4.2 (http://geas.dei.unipd.it/jdk1.4.2/docs/api/ https://www2.cs.duke.edu/csed/java/jdk1.4.2/docs/api/index.html).

In particolare, la libreria contiene classi che fanno uso della classe Stack, e quindi delle interfacce ad essa connesse (Collection, List, Iterator, ListIterator) del Java2 Collections Framework versione 1.4.2.

Sviluppate l'***adapter*** per la classe Stack utilizzando come ***adaptee*** la classe Vector di CLDC 1.1.

Lavorate in ambiente Java versione corrente (JDK 18), ma ricordate che e' FONDAMENTALE che il vostro codice utilizzi solo le funzionalita' presenti in CLDC 1.1 per realizzare l'adapter. Il vostro progetto DEVE poter essere compilato ed eseguito da linea di comando con il JDK 18.

Per evitare collisioni con le interfacce Collection, List, Iterator e ListIterator della versione corrente di Java ***dovete*** definire localmente al package del vostro adapter (package che ***dovete*** chiamare *myAdapter* senza ulteriori livelli di nidificazione) le interfacce HList, HCollection, HIterator ed HListIterator con tutti i metodi delle interfacce List, Collection, Iterator e ListIterator della versione 1.4.2 di Java completamente funzionanti. La vostra classe adapter ***deve*** chiamarsi StackAdaper, ***deve*** appartenere al package myAdapter e ***deve*** implementare le interfacce HList, HCollection.

Il comportamento dei vostri adapters e dei loro metodi ***deve*** essere esattamente quello descritto dalla documentazione di J2SE 1.4.2 e ***devono*** essere implementate tutte le optional operations.

Devono essere compliant con la documentazione della versione 1.4.2 anche gli iteratori e ***devono*** essere implementate anche tutte le optional operations degli iteratori stessi. La o le classi che implementano gli iteratori ***devono*** far parte del package myAdapter e ***deve/devono*** implementare le interfacce HIterator e HListIterator.

Dovete utilizzare la metodologia **Test Driven Development**, e, quindi, definire ed Implementare le test suite Junit per le classi sviluppate. Le classi di test ***devono*** essere contenute in un package myTest (senza ulteriori livelli di nidificazione). Il package deve contenere una classe TestRunner che possa essere invocata da linea di comando, eseguire tutti i test da voi definiti, fornire il risultato dei test ed il numero complessivo di test eseguiti e/o falliti.

Documentate la/le vostra test suite utilizzando il template "SAFe" descritto nella tabella 1 di questo documento.

Documentate ogni test case secondo il template "Homework" descritto nella tabella 2 di questo documento.

E' possibile fornire la documentazione in formato pdf o in formato javadoc, in entrambi i casi le diverse voci per i diversi test devono essere facilmente leggibili e distinguibili dalle altre.

Utilizzate il framework Junit nella versione usata a lezione. Dovete comunque dichiarare nella documentazione la versione utilizzata, le componenti del framework utilizzate e le eventuali librerie di matcher utilizzate. Se utilizzerete librerie matcher, esse devono essere fornite nella consegna in formato jar, in una cartella dedicata denominata Matcher.

Scrivete la documentazione delle classi (utilizzate il tool javadoc) fornendo almeno la descrizione delle classi e la documentazione di base (paragrafi parameters, returns, throws) dei metodi. Non vi e' impedito l'uso di annotazioni avanzate, ma non e' obbligatorio.

Tabella 1

Test suite section	Section description	Basic test suite template	Default test suite template	SAFe test suite template
Summary	A detailed description of the test suite. In this section, you can also set up test suite categories to help you organize your test suite into logical groups. This section uses a full-text editor.	X	X	X
Test Suite Design	A description of the design of the test suite.	X	X	X
Formal Review	A formal review process that can help your business processes comply with applicable industry standards and regulations.		X	
Pre-Condition	The information that must be true before the test suite is run. This section uses a full-text editor.		X	X
Post-Condition	The information that must be true after the test suite is run. This section uses a full-text editor.		X	X
Expected Results	The conditions that must be met before a test suite is considered to be successful.		X	

Test suite section	Section description	Basic test suite template	Default test suite template	SAFe test suite template
	This section uses a full-text editor.			
Risk Assessment	The risks that are associated with the test suite.		X	
Test Cases	In this section, you can add test cases, associate test environments, and run the test cases.	X	X	X
Test Suite Execution Records	The test suite execution records that are associated with the test suite.	X	X	X
Attachments	The files and documents that are attached to the test suite, such as screen captures and other supporting material.	X	X	
Execution Variables	The execution variables for manual test scripts. You can set the execution variable values in this section and then pass those values to the manual test scripts during run time. When you create a manual test script, add variables with the same names to the Execution Variables view in the Manual Test editor.		X	X

Tabella 2

Test case section	Section description	Homework test case template	Basic test case template	Default test case template	Collaborative ALM test case template	Agile test case template	Software test Description Test Case Template	Classic Test Case Template	SAFe test case template
Summary	A detailed description of the test case. In this section, you can also set up categories to organize your test cases into logical groups. This section uses a full-text editor.	X	X	X	X	X	X	X	X
Test Case Scope	The software and system for the test case. In this section, include an overview of the purpose, security, or privacy implications of the software or system. This section uses a full-text editor.						X		
Test Case Design	The overall design of the test case, including any background setup information or topologies. You can generate manual test scripts automatically from this content. This section uses a full-text editor.	X		X		X	X	X	X
Normative and Informative Documents	Attached or linked normative and informative documents, such as						X		

Test case section	Section description	Homework test case template	Basic test case template	Default test case template	Collaborative ALM test case template	Agile test case template	Software test Description Test Case Template	Classic Test Case Template	SAFe test case template
	applicable standards or guidelines specific to an industry or organization.								
Formal Review	A formal review process that can help your business processes comply with applicable industry standards and regulations.			X			X	X	
Development Items	Work items, such as defects and plan items that you submit in Rational Team Concert™.			X	X	X	X	X	X
Requirements	The requirements for a test cycle.		X	X			X	X	
Requirements Links	The requirements that link to external applications.			X	X	X	X		X
Risk Assessment	The risks that are associated with the test case.			X			X	X	
Test Preparation	The test-preparation details for the test, such as the preparations, procedures, or personnel actions that are required to run the test. In this section, include the procedures necessary to						X		

Test case section	Section description	Homework test case template	Basic test case template	Default test case template	Collaborative ALM test case template	Agile test case template	Software test Description Test Case Template	Classic Test Case Template	SAFe test case template
	prepare or configure the hardware or software that is used in the test. This section uses a full-text editor.								
Test Description	The description of the test case. The description might include a project-unique identifier of the test case, prerequisites for running the test, a description of the test data, an explanation of the criteria to evaluate the results with, and any assumptions or constraints that are associated with the test case. This section uses a full-text editor.	X					X		
Pre-Condition	The information that must be true before you run the test case. This section uses a full-text editor.	X		X			X	X	X
Post-Condition	The information that must be true after you run the test case. This section uses a full-text editor.	X		X			X	X	X

Test case section	Section description	Homework test case template	Basic test case template	Default test case template	Collaborative ALM test case template	Agile test case template	Software test Description Test Case Template	Classic Test Case Template	SAFe test case template
Expected Results	The conditions that must be met before a test case is considered to be successful. This section uses a full-text editor.	X		X			X	X	
Test Scripts	The test scripts that are associated with a test case.		X	X	X	X	X	X	X
Test Case Execution Records	The execution records for a test case.			X	X	X	X	X	X
Attachments	The files and documents that are attached to the test case, such as screen captures and other supporting material.		X	X			X	X	
Execution Variables	The execution variables for manual test scripts. You can set the execution variable values and then pass those values to the manual test scripts during run time. When you create a manual test script, add variables with the same names to the Execution Variables view in the Manual Test editor.			X				X	X