Quiz navigation

Exam sheet

1

Theoretical questions

2 3 4 5 6 7 8 9
10 11 12 13 14 15 16 17
18 19 20 21 22 23 24 25

Practical exercises

26 27 28 29 30

Show one page at a time

Finish review

Started on	Tuesday, 19 December 2023, 2:12 PM
State	Finished
Completed on	Tuesday, 19 December 2023, 3:07 PM
Time taken	54 mins 58 secs
Grade	**25.55** out of 40.00 (**63.88**%)
Feedback	**You have reached the minimum required points.**

Question **1**
Not answered
Not graded
Flag question

Below is the student exam sheet. If the information contained therein is not accurate, please inform the exam supervisor.
When the exam supervisor verifies your identity, please show the examination sheet to them!
This is not a task; there are no further actions related to this!

Answer:

Last name: Mean

First name: Diamand

Neptun: KSG25Z

Moodle ID: 39388

Question **2**
Incorrect
Mark 0.00 out of 1.00
Flag question

Which statement(s) is/are true about the Kanban method? Choose **all true** statements.
There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] Its disadvantage is that it can only display one project at a time.
- [ ] Its disadvantage is that it requires manual intervention by the Scrum master to keep it up-to-date.
- [x] It provides an overview of the status of each task according to the elements of the lifecycle.
- [ ] It cannot be used in Scrum-based development projects because the methodologies require different steps.

Your answer is incorrect.
The correct answer is:
It provides an overview of the status of each task according to the elements of the lifecycle.

Question **3**
Correct
Mark 1.00 out of 1.00
Flag question

Which of the following statements are true about *centralised* version control systems? Choose **all true** statements.
There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [ ] They must always manage multiple parallel branches.
- [ ] Any local modification (from a user's point of view) is fast, as it does not need to communicate with other actors.
- [ ] Normally, each user has the full repository, which contains all files with all versions and all metadata.
- [x] They may use locking to avoid conflicts.

Your answer is correct.
The correct answer is:
They may use locking to avoid conflicts.

Question **4**
Incorrect
Mark 0.00 out of 1.00
Flag question

We have provided 4 statements related to the quality of the source code. Choose **all true** statements!
There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [ ] Checklists are not necessary to be used for manual code review if source code version control is done with the Git tool.
- [ ] Manual code review should be done after running the automatic tools.
- [x] It may be useful to check our source code with several static checkers.
- [x] There are tools that can automatically check compliance with coding guidelines (e.g. MISRA C).

Your answer is incorrect.
The correct answers are:
There are tools that can automatically check compliance with coding guidelines (e.g. MISRA C).,
It may be useful to check our source code with several static checkers.,
Manual code review should be done after running the automatic tools.

Question **5**
Incorrect
Mark -0.25 out of 1.00
Flag question

We have provided 4 statements related to the behaviour modelled by the below state machine. Choose the true one.
A correct answer is worth +1, an incorrect answer -0.25 and a blank answer 0 points.

- ( ) The *stop* behaviour is always triggered by the *exitProcess* event.
- ( ) It is possible that the *turnOn* event will cause the *initialize* behaviour to be executed but the system is put in the *Idle* state.
- (•) The *initialize* event always puts the system in the *Working* state.
- ( ) The *turnOff* event triggers the *stop* event.

Your answer is incorrect.
The correct answer is:
It is possible that the *turnOn* event will cause the *initialize* behaviour to be executed but the system is put in the *Idle* state.

Question **6**
Correct
Mark 1.00 out of 1.00
Flag question

Which of the following ones are typical software development lifecycle tasks?
There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] Plan
- [x] Test
- [x] Code

☑ Release

Your answer is correct.

The correct answers are:
Plan,

Code,

Test,

Release

Question **7**
Incorrect
Mark 0.00 out of 1.00
Flag question

Which of the following ones are part of the design phase of software development?

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- ☑ Documenting decisions
- ☑ Transforming the requirements into use cases and processes
- ☐ Collecting requirements
- ☐ Evaluation of alternatives

Your answer is incorrect.

The correct answers are:
Evaluation of alternatives,

Documenting decisions,

Transforming the requirements into use cases and processes

Question **8**
Partially correct
Mark 0.80 out of 1.00
Flag question

Which activity is specific to which (requirement) checking method? Match the corresponding ones. It is possible that some of the elements do not have a counterpart.

A correct answer is worth +0.2 points.

Manually reading and analysing documents, executed by humans	Review
Checking feasibility	Prototypes
Using structured check lists	Review
Connecting requirements, examples, and tests	User story
Better understanding of requirements through examples	Specification by example

Your answer is partially correct.

You have correctly selected 4.
The correct answer is:
Manually reading and analysing documents, executed by humans → Review,

Checking feasibility → Prototypes,

Using structured check lists → Review,

Connecting requirements, examples, and tests → Specification by example,

Better understanding of requirements through examples → Specification by example

Question **9**
Incorrect
Mark 0.00 out of 1.00
Flag question

We have provided 4 statements related to the syntax and semantics of the modelling languages. Choose **all true** statements!

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- ☐ The abstract syntax defines the possible element and relation types.
- ☑ The semantics of the model mainly focuses on the meaning of the model.
- ☑ The concrete syntax may contain a rule like "the model may contain classes".
- ☐ The precise definition of the semantics is typically not more difficult than defining the syntax.

Your answer is incorrect.

The correct answers are:
The abstract syntax defines the possible element and relation types.,

The semantics of the model mainly focuses on the meaning of the model.

Question **10**
Incorrect
Mark 0.00 out of 1.00
Flag question

Which statements are true for software?

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- ☐ Generally, during the development, it is easier to repair a fault in a software component than one in a hardware component.
- ☑ It is easy to "produce" a new instance of a software.
- ☐ In case of a software component, we have to expect the same kind of faults as in case of hardware components.
- ☐ The software does not have to deal with any constraints of the physical world.

Válasza helytelen.

The correct answers are:
It is easy to "produce" a new instance of a software.,

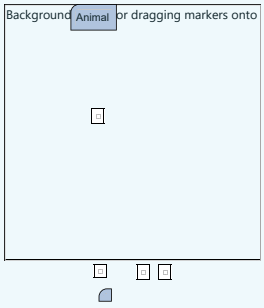
Generally, during the development, it is easier to repair a fault in a software component than one in a hardware component.

Question **11**
Correct
Mark 1.00 out of 1.00
Flag question

We want to model that the abstract class Animal is the generalization of the class Cat.

Complete the model by pulling the right relation and the right class name to the right position.

A correct answer is worth +0.5 points.



Your answer is correct.

Question **12**
Correct
Mark 1.00 out of 1.00
Flag question

At which level of the review are the following participants and moderators typical?

A correct answer is worth +0.25 points.

Involving external experts, led by a specially trained moderator	Inspection
Involving external experts, led by a team member	Technical review
Teammates, led by the author	Walk through
Teammates, possibly the team leader, without a moderator	Informal review

Your answer is correct.

The correct answer is:
Involving external experts, led by a specially trained moderator → Inspection,

Involving external experts, led by a team member → Technical review,

Teammates, led by the author → Walk through,

Teammates, possibly the team leader, without a moderator → Informal review

Question **13**
Incorrect
Mark 0.00 out of 1.00
Flag question

If there is an already existing code base, which ones of the following operations are considered as "*refactor*" operations?

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] Extracting some instructions from the body of an existing method into a new one.
- [ ] Extending the functionality of a method.
- [ ] Renaming a method.
- [ ] Renaming a variable.

Your answer is incorrect.

The correct answers are:
Extracting some instructions from the body of an existing method into a new one.,

Renaming a variable.,

Renaming a method.

Question **14**
Correct
Mark 1.00 out of 1.00
Flag question

Which version control-related concept can be defined as: a representation of related changes and the meta-data associated to them?

Write one single word as an answer.

Answer: commit

The correct answer is: Commit

Question **15**
Incorrect
Mark 0.00 out of 1.00
Flag question

Which statement(s) is/are true about scheduling in a development project following the Scrum method? Choose **all true** statements.

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [ ] The results of a *Sprint* do not need to be reviewed in a separate meeting, as they are accurately tracked during the *Daily standup* meetings.
- [ ] The purpose of the *Sprint retrospective* meeting is to plan the next week's work.
- [x] During *Sprint planning*, the tasks for the next period are selected from the items in the *Product backlog* list.
- [ ] The purpose of the *Daily standup* meeting is to plan the next day's work and identify potential obstacles.

Your answer is incorrect.

The correct answers are:
The purpose of the *Daily standup* meeting is to plan the next day's work and identify potential obstacles.,

During *Sprint planning*, the tasks for the next period are selected from the items in the *Product backlog* list.

Question **16**
Correct
Mark 1.00 out of 1.00
Flag question

Choose **all true** statements.

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] Black box testing is a specification-based technique.
- [ ] SUT is the abbreviation of *Simple Unit Test*.
- [x] We choose one test input from each equivalence class.
- [ ] We usually do not check the so called main path (happy path).

Your answer is correct.

The correct answers are:
Black box testing is a specification-based technique.,

We choose one test input from each equivalence class.

Question **17**
Correct
Mark 1.00 out of 1.00
Flag question

When executing a test case, one of the asserts in it evaluates to false. What will be the result of the test?

A correct answer is worth +1, an incorrect answer -0.25 and a blank answer 0 points.

- (•) fail
- ( ) inconclusive
- ( ) error
- ( ) pass

Válasza helyes.

The correct answer is:
fail

Question **18**
Incorrect
Mark 0.00 out of 1.00
Flag question

We have provided 4 statements related to structure-based test design. Choose **all true** statements!

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] A decision may have more than two possible outcomes.
- [x] A possible example of a *statement* is to assign a value to a variable (e.g. int x = 10;).
- [ ] The 100% statement coverage implies the 100% decision coverage.
- [ ] If the statement covergae is 100%, this also guarantees the coverage of all non-empty branches.

Your answer is incorrect.

The correct answers are:
If the statement covergae is 100%, this also guarantees the coverage of all non-empty branches.,

A decision may have more than two possible outcomes.,

A possible example of a *statement* is to assign a value to a variable (e.g. int x = 10;).

Question **19**
Correct
Mark 1.00 out of 1.00
Flag question

Which of the following ones are called functional requirements?

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [ ] Requirements for the ergonomics of the user interface
- [ ] Requirements for the number of requests that can be processed per second
- [ ] Requirements for the expected failure frequency of the system
- [x] Requirements for which operations the user can perform

Your answer is correct.

The correct answer is:
Requirements for which operations the user can perform

Question **20**
Correct
Mark 1.00 out of 1.00
Flag question

By software we only mean the source code.

A correct answer is worth +1, an incorrect answer -0.5 and a blank answer 0 points.

- ( ) True
- (•) False

Válasza helyes.

The correct answer is:
False

Question **21**
Incorrect
Mark 0.00 out of 1.00
Flag question

Pull the missing words to the right places in the following statements.

A correct answer is worth +0.25 points.

The [ ] blocks the initiating party.

The [ ] of a message happens always after its [all messages] - this is called [ ] .

asynchronous message | reception | orderedness | causality | sending | synchronous call

Your answer is incorrect.

The correct answer is:
Pull the missing words to the right places in the following statements.

A correct answer is worth +0.25 points.

The [synchronous call] blocks the initiating party.

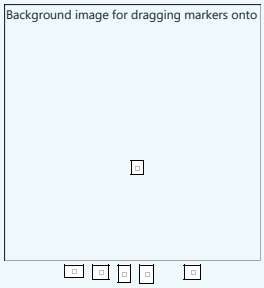
The [reception] of a message happens always after its [sending] - this is called [causality] .

Question **22**
Incorrect
Mark 0.00 out of 1.00
Flag question

We want to model that a wall may contain windows, and there is a whole-part relation between them, i.e. one window can only be contained by one wall, but not more.

Choose the right relation and pull it between the classes!

A correct answer is worth +1 point.



Your answer is incorrect.

Question **23**
Correct
Mark 1.00 out of 1.00
Flag question

Which of the following statements are true about *distributed* version control systems? Choose **all true** statements.

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [x] They must always be able to manage multiple parallel branches.
- [ ] They usually use locking to avoid conflicts.
- [x] Normally, each user has the full repository, which contains all files with all versions and all metadata.
- [x] Any local modification (from a user's point of view) is fast, as it does not need to communicate with other actors.

Your answer is correct.

The correct answers are:
Normally, each user has the full repository, which contains all files with all versions and all metadata.,

Any local modification (from a user's point of view) is fast, as it does not need to communicate with other actors.,

They must always be able to manage multiple parallel branches.

Question **24**
Correct
Mark 1.00 out of 1.00
Flag question

Which statement(s) is/are true about the person playing the role of the *Product Owner* in an agile software development project? Choose **all true** statements.

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

- [ ] He or she is responsible for the internal meetings that are part of the Scrum process.
- [x] He or she maintains the *Product Backlog* list.
- [x] It is primarily their responsibility is to communicate with all stakeholders.
- [ ] He or she prepares an accurate schedule of the project members' tasks.

Your answer is correct.

The correct answers are:
It is primarily their responsibility is to communicate with all stakeholders.,

He or she maintains the *Product Backlog* list.

Question **25**
Incorrect
Mark 0.00 out of

Which statement(s) is/are true for software life cycle models? Choose **all true** statements.

There can be more than one correct answer. Points are awarded only if all correct answers are marked!

1.00

Flag question

- [x] A typical goal in agile software development is the early and continuous delivery of software.
- [x] In software development projects following the waterfall model, the individual phases cannot be checked because the software to be delivered is only completed in the last step.
- [ ] The waterfall model is useful for complex, longer projects because the tasks are well separated from each other.
- [ ] In the V-model, the traceability of the development is ensured by the frequent release of the software.

Your answer is incorrect.

The correct answer is:
A typical goal in agile software development is the early and continuous delivery of software.

Question **26**

Correct

Mark 1.00 out of 1.00

Flag question

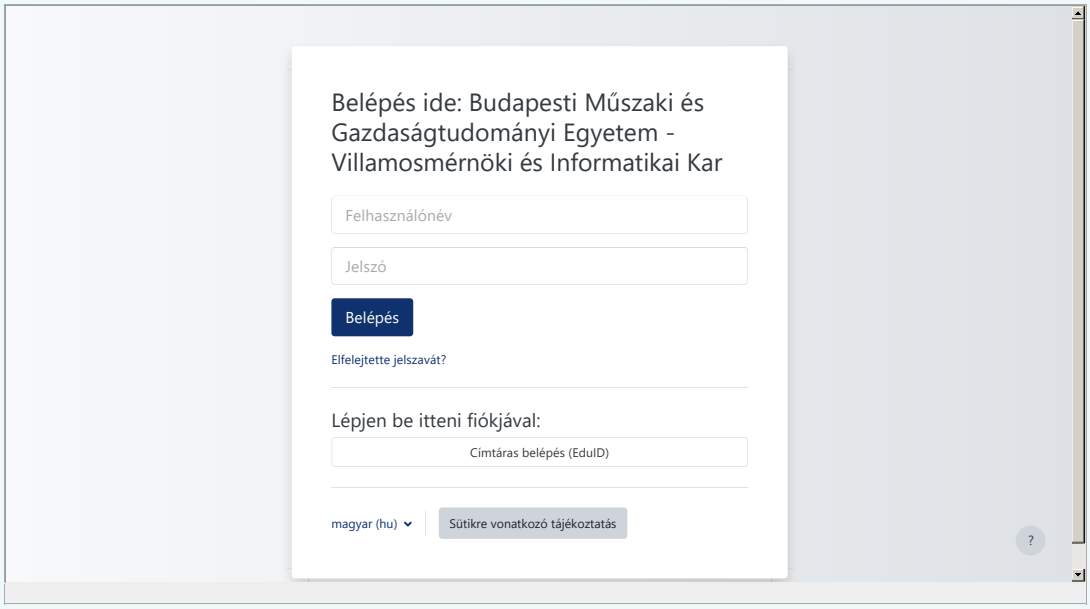
Which (exactly one) Git command must be issued to get your repository from its current state to the desired target state?

Get your repository to the desired target state by issuing Git commands.

There are no sub-points for this task, the score is awarded only for the correct solution. If you have done something wrong, use the "Reset answer" button, or use the "undo" command to undo the last command.

Answer: (penalty regime: 0 %)

Reset answer



Expected	Got

<pre>{
  "branches": {
  "main": {
  "target": "C4",
  "id": "main",
  "remoteTrackingBranchID": null
  },
  "feature": {
  "target": "C2",
  "id": "feature",
  "remoteTrackingBranchID": null
  }
  },
  "commits": {
  "C0": {
  "parents": [],
  "id": "C0",
  "rootCommit": true
  },
  "C1": {
  "parents": [
  "C0"
  ],
  "id": "C1"
  },
  "C2": {
  "parents": [
  "C1"
  ],
  "id": "C2"
  },
  "C3": {
  "parents": [
  "C1"
  ],
  "id": "C3"
  },
  "C4": {
  "parents": [
  "C2",
  "C3"
  ],
  "id": "C4"
  }
  },
  "tags": {},
  "HEAD": {
  "target": "main",
  "id": "HEAD"
  }
}</pre>
	<pre>{
	  "branches": {
	  "main": {
	  "target": "C4",
	  "id": "main",
	  "remoteTrackingBranchID": null
	  },
	  "feature": {
	  "target": "C2",
	  "id": "feature",
	  "remoteTrackingBranchID": null
	  }
	  },
	  "commits": {
	  "C0": {
	  "parents": [],
	  "id": "C0",
	  "rootCommit": true
	  },
	  "C1": {
	  "parents": [
	  "C0"
	  ],
	  "id": "C1"
	  },
	  "C2": {
	  "parents": [
	  "C1"
	  ],
	  "id": "C2"
	  },
	  "C3": {
	  "parents": [
	  "C1"
	  ],
	  "id": "C3"
	  },
	  "C4": {
	  "parents": [
	  "C2",
	  "C3"
	  ],
	  "id": "C4"
	  }
	  },
	  "tags": {},
	  "HEAD": {
	  "target": "main",
	  "id": "HEAD"
	  }
	}</pre>

Passed all tests!

▸ Show/hide question author's solution (Nodejs)

Correct

Marks for this submission: 1.00/1.00.

Question **27**

Correct

Mark 3.00 out of 3.00

Flag question

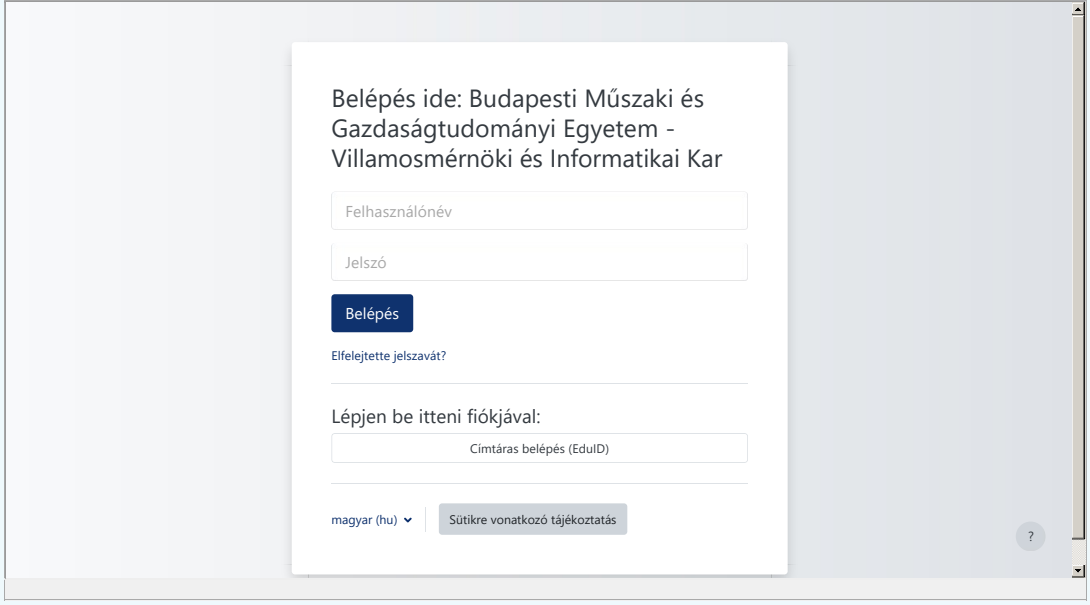
Pete was working on two features at the same time (*feature-a*, *feature-b*). However, at the daily stand-up, it was decided that these two features were so similar that they should be developed together. Pete was given the task to integrate *feature-b* with *feature-a*, but in such a way that the version history remains linear.

Get your repository to the desired target state by issuing Git commands.

There are no sub-points for this task, the score is awarded only for the correct solution. If you have done something wrong, use the "Reset answer" button, or use the "undo" command to undo the last command.

Answer: (penalty regime: 0 %)

Reset answer



Expected	Got

{
"branches": {
"main": {
"target": "C2",
"id": "main",
"remoteTrackingBranchID": null
},
"feature-a": {
"target": "C3'",
"id": "feature-a",
"remoteTrackingBranchID": null
},
"feature-b": {
"target": "C4",
"id": "feature-b",
"remoteTrackingBranchID": null
}
},
"commits": {
"C0": {
"parents": [],
"id": "C0",
"rootCommit": true
},
"C1": {
"parents": [
"C0"
],
"id": "C1"
},
"C2": {
"parents": [
"C1"
],
"id": "C2"
},
"C3": {
"parents": [
"C1"
],
"id": "C3"
},
"C4": {
"parents": [
"C1"
],
"id": "C4"
},
"C3'": {
"parents": [
"C4"
],
"id": "C3'"
}
},
"tags": {},
"HEAD": {
"target": "feature-a",
"id": "HEAD"
}
}
	{
	"branches": {
	"main": {
	"target": "C2",
	"id": "main",
	"remoteTrackingBranchID": null
	},
	"feature-a": {
	"target": "C3'",
	"id": "feature-a",
	"remoteTrackingBranchID": null
	},
	"feature-b": {
	"target": "C4",
	"id": "feature-b",
	"remoteTrackingBranchID": null
	}
	},
	"commits": {
	"C0": {
	"parents": [],
	"id": "C0",
	"rootCommit": true
	},
	"C1": {
	"parents": [
	"C0"
	],
	"id": "C1"
	},
	"C2": {
	"parents": [
	"C1"
	],
	"id": "C2"
	},
	"C3": {
	"parents": [
	"C1"
	],
	"id": "C3"
	},
	"C4": {
	"parents": [
	"C1"
	],
	"id": "C4"
	},
	"C3'": {
	"parents": [
	"C4"
	],
	"id": "C3'"
	}
	},
	"tags": {},
	"HEAD": {
	"target": "feature-a",
	"id": "HEAD"
	}
	}

Passed all tests!

▸ Show/hide question author's solution (Nodejs)

Correct

Marks for this submission: 3.00/3.00.

Question **28**

Correct

Mark 4.00 out of 4.00

Flag question

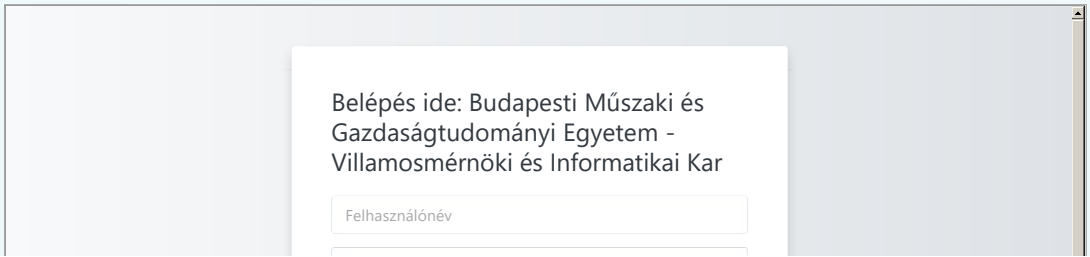
Alice and Bob are working together on the new feature (*feature-a*). Both Alice and Bob did their part, but Alice was the quicker one, she uploaded her changes to the remote repository first. Now Bob wants to share his work with the team.

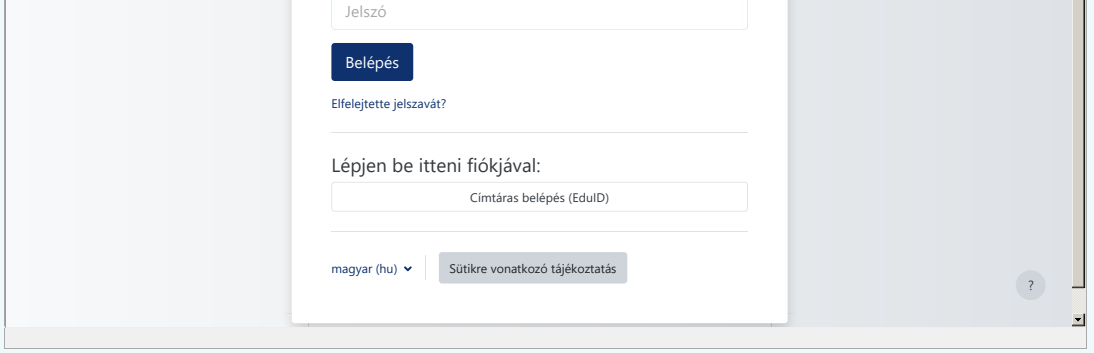
Get your repository to the desired target state by issuing Git commands.

There are no sub-points for this task, the score is awarded only for the correct solution. If you have done something wrong, use the "Reset answer" button, or use the "undo" command to undo the last command.

Answer: (penalty regime: 0 %)

Reset answer





Expected	Got

{
"branches": {
"main": {
"target": "C2",
"id": "main",
"remoteTrackingBranchID": "o/main"
},
"feature-a": {
"target": "C6",
"id": "feature-a",
"remoteTrackingBranchID": "o/feature-a"
},
"o/main": {
"target": "C2",
"id": "o/main",
"remoteTrackingBranchID": null
},
"o/feature-a": {
"target": "C6",
"id": "o/feature-a",
"remoteTrackingBranchID": null
}
},
"commits": {
"C0": {
"parents": [],
"id": "C0",
"rootCommit": true
},
"C1": {
"parents": [
"C0"
],
"id": "C1"
},
"C2": {
"parents": [
"C1"
],
"id": "C2"
},
"C3": {
"parents": [
"C1"
],
"id": "C3"
},
"C5": {
"parents": [
"C3"
],
"id": "C5"
},
"C4": {
"parents": [
"C3"
],
"id": "C4"
},
"C6": {
"parents": [
"C4",
"C5"
],
"id": "C6"
}
},
"tags": {},
"HEAD": {
"target": "feature-a",
"id": "HEAD"
},
"originTree": {
"branches": {
"main": {
"target": "C2",
"id": "main",
"remoteTrackingBranchID": null
},
"feature-a": {
"target": "C6",
"id": "feature-a",
"remoteTrackingBranchID": null
}
},
"commits": {
"C0": {
"parents": [],
"id": "C0",
"rootCommit": true
},
"C1": {
"parents": [
"C0"
],
"id": "C1"
},
"C2": {
"parents": [
"C1"
],
"id": "C2"
},
"C3": {
"parents": [
"C1"
],
"id": "C3"
},
"C4": {
"parents": [
"C3"
],
"id": "C4"
},
	{
	"branches": {
	"main": {
	"target": "C2",
	"id": "main",
	"remoteTrackingBranchID": "o/main"
	},
	"feature-a": {
	"target": "C6",
	"id": "feature-a",
	"remoteTrackingBranchID": "o/feature-a"
	},
	"o/main": {
	"target": "C2",
	"id": "o/main",
	"remoteTrackingBranchID": null
	},
	"o/feature-a": {
	"target": "C6",
	"id": "o/feature-a",
	"remoteTrackingBranchID": null
	}
	},
	"commits": {
	"C0": {
	"parents": [],
	"id": "C0",
	"rootCommit": true
	},
	"C1": {
	"parents": [
	"C0"
	],
	"id": "C1"
	},
	"C2": {
	"parents": [
	"C1"
	],
	"id": "C2"
	},
	"C3": {
	"parents": [
	"C1"
	],
	"id": "C3"
	},
	"C5": {
	"parents": [
	"C3"
	],
	"id": "C5"
	},
	"C4": {
	"parents": [
	"C3"
	],
	"id": "C4"
	},
	"C6": {
	"parents": [
	"C4",
	"C5"
	],
	"id": "C6"
	}
	},
	"tags": {},
	"HEAD": {
	"target": "feature-a",
	"id": "HEAD"
	},
	"originTree": {
	"branches": {
	"main": {
	"target": "C2",
	"id": "main",
	"remoteTrackingBranchID": null
	},
	"feature-a": {
	"target": "C6",
	"id": "feature-a",
	"remoteTrackingBranchID": null
	}
	},
	"commits": {
	"C0": {
	"parents": [],
	"id": "C0",
	"rootCommit": true
	},
	"C1": {
	"parents": [
	"C0"
	],
	"id": "C1"
	},
	"C2": {
	"parents": [
	"C1"
	],
	"id": "C2"
	},
	"C3": {
	"parents": [
	"C1"
	],
	"id": "C3"
	},
	"C4": {
	"parents": [
	"C3"
	],
	"id": "C4"
	},

Expected	Got

<pre>"C5": {
  "parents": [
  "C3"
  ],
  "id": "C5"
  },
  "C6": {
  "parents": [
  "C4",
  "C5"
  ],
  "id": "C6"
  }
  },
  "tags": {},
  "HEAD": {
  "target": "feature-a",
  "id": "HEAD"
  }
  }
}</pre>
	<pre>"C5": {
	  "parents": [
	  "C3"
	  ],
	  "id": "C5"
	  },
	  "C6": {
	  "parents": [
	  "C4",
	  "C5"
	  ],
	  "id": "C6"
	  }
	  },
	  "tags": {},
	  "HEAD": {
	  "target": "feature-a",
	  "id": "HEAD"
	  }
	  }
	}</pre>

Passed all tests!

▸ Show/hide question author's solution (Nodejs)

Correct

Marks for this submission: 4.00/4.00.

Question **29**

Partially correct

Mark 2.00 out of 4.00

Flag question

Pattern-based Static Analysis

```
1 class A {
2   public int foo() {
3     return 0;
4   }
5 }
6
7 class B extends A {
8
9   String name = null;
10
11  public B(){
12    if(name.length() == 0) name = "ABC";
13  }
14
15  @Override
16  public int foo(int i) {
17    int[] arr = {1, 2, 3};
18    return arr[2];
19  }
20
21  public void bar(){
22    System.out.println(name);
23  }
24
25  int pow(int num, int exponent) {
26    if(exponent <= 0) return 1;
27    return num * pow(num, exponent - 1);
28  }
29 }
```

We have run a static checker on the Java source code shown above. The tool has reported several errors, decide which ones are real.

A correct answer is worth +1 point.

1) Does the following warning indicate a real error?
Line 18 - Potential ArrayIndexOutOfBoundsException
Explanation: Out-of-bounds indexing of an array happens when an element of the array is accessed, whose index lies outside of the valid range of the array.

The indicated warning is a false positive

Correct answer, well done.

2) Does the following warning indicate a real error?
Line 12 - Potential NullPointerException
Explanation: Using a pointer whose value is Null.

The indicated warning is a true positive

Correct answer, well done.

3) Does the following warning indicate a real error?
Lines 15-19 - The method does not override any methods of the parent classes
Explanation: With the @Override annotation the intention can be shown that the given method overrides a method of some of its parent classes. When this annotation is put on a method but it does not override any methods, the compilers and static analysis tools have to indicate an error.

The indicated warning is a false positive

Incorrect answer.

4) Does the following warning indicate a real error?
Lines 25-28 - Infinite recursion
Explanation: If a recursive method does not contain any means to break the recursion and return, the execution continues until the stack overflows and the program crashes.

The indicated warning is a true negative

Incorrect answer.

A correct answer is: `"false positive"`

A correct answer is: `"true positive"`

A correct answer is: `"true positive"`

A correct answer is: `"false positive"`

Question **30**

Correct

Mark 4.00 out of 4.00

Flag question

The below code fragment is given:

```
1 int retro(int x, double b, int d) {
2   if(x % 2 == 0) {
3     d = -d;
4   }
5
6   if(b < 3.0) {
7     return d;
8   }
9   else {
10    return 2 * d;
11  }
12 }
```

We have run a test suite on this code fragment, and got the JaCoCo coverage report shown above. Reminder:

- Green line: The instruction or decision is covered by the test suite.
- Yellow line: The decision is only partially covered by the test suite, there are decision outcomes that are uncovered.
- Red line: The instruction or decision is NOT covered by the test suite.

A correct answer is worth +2 points (per part).

a) Add a new test case to complete the existing test suite to reach 100% instruction coverage:

- x = 3 | 3
- b = 5 | 5
- d = 1 | 1

Correct answer, well done.

b) What decision coverage is reached by the completed test suite (the original one completed with your test case in **part a)**)?

The answer to the following question should be given as a rational fraction; if the answer is $\frac{2}{6}$, write 2/6 into the field! Do not simplify the fraction.

4/4 | $\frac{4}{4}$

Correct answer, well done.

A correct answer is 0, which can be typed in as follows: `0`

A correct answer is 3.0, which can be typed in as follows: `3.0`

A correct answer is 0, which can be typed in as follows: `0`

A correct answer is $\frac{3}{4}$, which can be typed in as follows: `3/4`

Finish review