

Durata task [Puncte]	Descriere task
2h	<ul style="list-style-type: none"> <li>Se va verifica modul de realizare a task-urilor primite în cadrul temei de laborator precedente, i.e., <b>Lab 03. Testare Black-Box</b>.</li> </ul>
2h	<ul style="list-style-type: none"> <li>Se va discuta proiectarea cazurilor de testare folosind criteriul white-box pentru metoda <code>cautaCarte</code> din proiectul <b>Biblioteca</b>, clasa <b>CartiRepoMockB</b>.</li> </ul> <pre> /**  * Construiește lista cartilor scrise de un autor dat  * @param autor - autorul pentru care se cauta cartile scrise  * @return lista cartilor scrise de autorul dat  */ @Override public List&lt;Carte&gt; cautaCarte(String autor) {     List&lt;Carte&gt; carti = getCarti();     List&lt;Carte&gt; cartiGasite = new ArrayList&lt;Carte&gt;();     int i=0;     while (i&lt;carti.size()){         boolean exista = false;         List&lt;String&gt; lautor = carti.get(i).getCuvinteCheie();         int j = 0;         exista = hasAutor(lautor, autor);         if(exista == true){             cartiGasite.add(carti.get(i));         }         i++;     }     return cartiGasite; }  private boolean hasAutor(List&lt;String&gt; autorList, String a) {     boolean exista = false;     for(String autor: autorList){         if(autor.toLowerCase().contains(a.toLowerCase())){             exista = true;             break;         }     }     return exista; } </pre>
<b>TEMĂ</b> [10 puncte]	<p>Se consideră următoarea variantă* a metodei <code>modifyEmployeeFunction</code> din proiectul <b>Angajați</b>, clasa <b>EmployeeMock</b>.</p> <pre> /**  * Modifica atributul 'functia didactica' pentru un angajat dat  * @param employee - angajatul pentru care se modifica  * atributul 'functia didactica'  * @param newFunction - noua functie didactica (ASISTENT,  * LECTOR, CONFERENTIAR, PROFESOR)  */ @Override public void modifyEmployeeFunction(Employee employee, DidacticFunction newFunction) {      if (employee!=null) {         int i = 0;         while (i &lt; employeeList.size()) {             if (employeeList.get(i).getId() == employee.getId())                 employeeList.get(i).setFunction(newFunction);             i++;         }     } } </pre>

	<div data-bbox="375 152 1433 241" style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <pre>         }     } } </pre> </div> <p>*) Dacă este necesar, se copiază codul de mai sus în proiectul <b>Angajați</b>.</p> <p>Realizați următoarele task-uri pentru metoda precizată:</p> <ol style="list-style-type: none"> <li>1. <b>[3 puncte]</b> Se vor elabora:             <ul style="list-style-type: none"> <li>○ Control Flow Graph (CFG) <b>[1 punct]</b>;</li> <li>○ calculul complexității ciclomatice (CC, 3 formule de calcul) <b>[1 punct]</b>;</li> <li>○ drumurile independente <b>[1 punct]</b>;</li> </ul> </li> <li>2. <b>[2 puncte]</b> Se vor proiecta cazuri de testare pentru:             <ol style="list-style-type: none"> <li>1. acoperirea instrucțiunilor (statement coverage, <b>sc</b>);</li> <li>2. acoperirea deciziilor/condițiilor (decision/condition coverage; <b>dc, cc, dcc</b>);</li> <li>3. acoperirea condițiilor multiple (multiple condition coverage, <b>mcc</b>);</li> <li>4. acoperirea drumurilor (all path coverage, <b>apc</b>);</li> <li>5. acoperirea ciclurilor simple / imbricate (simple and nested loop coverage, <b>lc</b>).</li> </ol> </li> <li>3. <b>[3 puncte]</b> Se vor implementa toate cazurile de testare proiectate anterior, folosind JUnit.</li> <li>4. <b>[2 puncte]</b> Se va determina gradul/procentul de acoperire cu teste rulate pentru metoda testată, i.e., metoda pentru care s-a elaborat CFG (vezi <b>Tutorial Coverage</b>). Valoarea obținută se va completa în fișierul <b>Lab04_WBT_TCs_Form.xlsx</b> în secțiunea <b>Statistics</b>.</li> </ol> <p>Pentru cerințele 1 --&gt; 4 se va folosi fișierul <b>Lab04_WBT_TCs_Form.xlsx</b>.</p> <p><b>Observație</b></p> <ul style="list-style-type: none"> <li>• În situația în care cazurile de testare proiectate și implementate evidențiază defecte (rezultatul așteptat nu este identic cu rezultatul obținut în urma rulării testelor), codul sursă se va depana și procesul de testare se va relua pentru toate cazurile de testare. <u>În situația modificării codului sursă, se reiau TOATE task-urile de testare.</u></li> </ul>
<b>[facultativ]</b>	<ul style="list-style-type: none"> <li>• Se va exemplifica proiectarea cazurilor de testare folosind criteriul white-box (vezi <b>MaxScoreQuizCounter_TCs_WBT.xlsx</b>) cu codul descris în fișierul <b>MetodaFN.pdf</b>.</li> </ul>