|  |  |  |
| --- | --- | --- |
| **ISTANBUL TECHNICAL UNIVERSITY FACULTY OF COMPUTER AND INFORMATICS** | | |
|  | **BLG475E - Software Quality and Testing  Final Exam Sample** | |
| **Date:** |  |
| **Duration:** | 120 mins. |
| **Instructors:** | Assist. Prof. Dr. Ayşe Tosun  Lecturers from Turkcell |
| **Notes:** | Closed book, dictionaries are allowed! Write your answers on this paper. |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Student ID:** |  | **Name, Surname:** |  | **Signature:** |  | | | |

**Q. 1: (15 points)** What is a “stub”? What is a “driver”? For which test strategies are they useful?

**Q. 2: (20 points)** Give an example of a program (write its code) that can be tested by complete (100%) statement coverage but NOT by complete (100%) branch coverage. Explain why you cannot achieve 100% branch coverage.

**Q. 3:** **(20 points)** Let us study the following program:

x=0;

y = function\_that\_retrieves\_y(); //call this funcY() in your graph.

while (y > 100)

{

x=x+y;

y = function\_that\_retrieves\_y();

}

if (y < 200)

print(x);

else

print(y);

1. Construct a control-flow graph for the program.
2. Design test cases for reaching 100% branch coverage over the program. Use as few test cases as possible.

**Q. 4:** **(18 points)** The program in Q3 has been modified into the following form, so it must be regression tested:

x=0;

y = function\_that\_retrieves\_y();

while (y > x)

{

x=x+y;

y = function\_that\_retrieves\_y();

}

if (x < 100)

print(x+x);

else

print(x);

1. Redraw the control flow graph.
2. List all def-use paths with respect to variable x.
3. Design test cases for covering the complete def-use paths with respect to variable x in the modified program. If possible, reuse test cases designed in the earlier question.

**Q. 5: (12 points)**

***Mean Time Between Failures (MTBF)*** is the predicted elapsed time between inherent failures of a system during operation. This measure typically assumes that the failed system is immediately repaired.

***Mean Time To Repair (MTTR)*** is the average time that a system will take to recover from any failure.

What is the relationship between these two metrics?

Which quality attribute(s) do you think could be measured with MTBF and MTTR? (Answer the question for each of the metrics separately.)

**Q.6: (15 points)** What is the objective of performance testing?

Who performs performance tests?

What are the most common issues detected during performance testing?