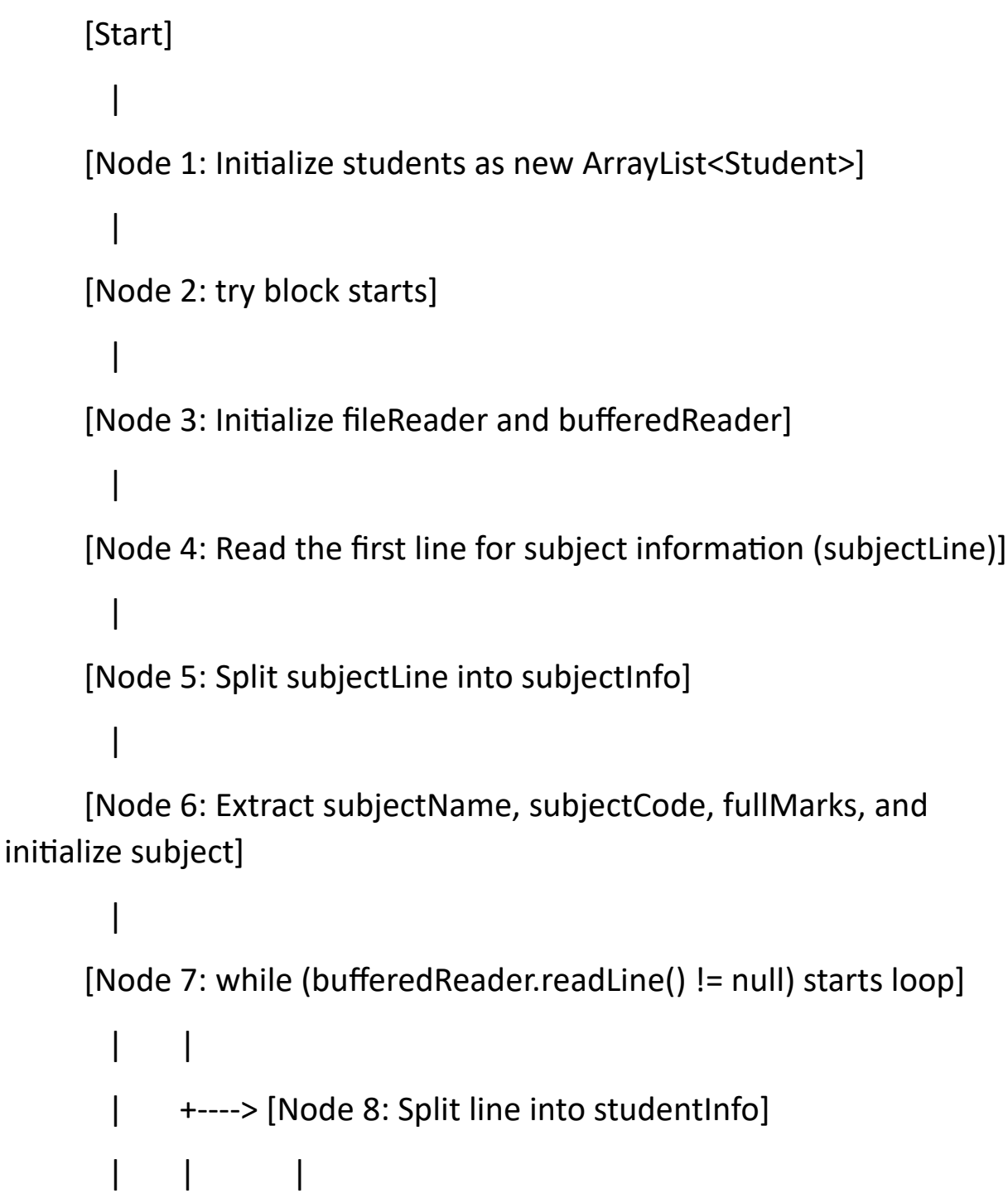
Data Flow Coverage

```
public FileProcessor(String fileName)
```

1. **Start** - Entry point of the **FileProcessor** constructor.
2. **Node 1** - Initialize **students** as a new **ArrayList<Student>**.
3. **Node 2** - **try** block starts.
4. **Node 3** - Initialize **fileReader** and **bufferedReader**.
5. **Node 4** - Read the first line for subject information (**subjectLine**).
6. **Node 5** - Split **subjectLine** into **subjectInfo**.
7. **Node 6** - Extract **subjectName**, **subjectCode**, **fullMarks**, and initialize **subject**.
8. **Node 7** - **while** loop starts to read student information (**line**).
    - **Node 8** - Inside the loop, split **line** into **studentInfo**.
    - **Node 9** - Extract student information and create a **Student** object.
    - **Node 10** - Add **Student** to **students**.
    - **Node 11** - End of loop body, go back to **Node 7**.
9. **Node 12** - After the loop, set **subject.enrolledStudents**.
10. **Node 13** - Close **bufferedReader**.
11. **Node 14** - **catch(FileNotFoundException e)** block.
12. **Node 15** - Print "File was not found!".
13. **Node 16** - **catch (Exception e)** block.
14. **Node 17** - Print "Error, Could not process the file!".

15. **End** - End of the constructor.

Control Flow Graph (CFG)



| | [Node 9: Extract student information and create a Student object]

| | |

| | [Node 10: Add Student to students]

| | |

| | [Node 11: Loop back to Node 7]

| |

| v

[Node 12: Set subject.enrolledStudents(students)]

|

[Node 13: Close bufferedReader]

|

v

[End]

|

[Node 14: catch(FileNotFoundException e)]

|

[Node 15: Print "File was not found!"]

|

[End]

|

[Node 16: catch(Exception e)]

|

[Node 17: Print "Error, Could not process the file!"]

|

[End]

Definitions (DEFs):

- **students** at **Node 1**.
- **fileReader** at **Node 3**.
- **bufferedReader** at **Node 3**.
- **subjectLine** at **Node 4**.
- **subjectInfo** at **Node 5**.
- **subjectName**, **subjectCode**, **fullMarks** at **Node 6**.
- **subject** at **Node 6**.
- **line** at **Node 7**.
- **studentInfo** at **Node 8**.
- **studentName**, **studentNumber**, **activitiesMark**, **practicalMark**, **midtermExamMark**, **finalExamMark** at **Node 9**.
- **student** at **Node 9**.
- **students** at **Node 10** (modified).
- **subject** at **Node 12** (modified with enrolled students).

Uses (USEs):

- **fileReader** at **Node 3** (for **BufferedReader** initialization).
- **bufferedReader** at **Node 4** (for reading **subjectLine**).

- **subjectLine** at **Node 5** (for splitting).
- **subjectInfo** at **Node 6** (for extracting **subjectName**, **subjectCode**, **fullMarks**).
- **bufferedReader** at **Node 7** (for reading **line**).
- **line** at **Node 8** (for splitting).
- **studentInfo** at **Node 9** (for extracting student data).
- **students** at **Node 12** (for setting enrolled students).

All-DEF, All-USE, and ADUP Analysis

All-DEF:

- **students** (Node 1)
- **fileReader** (Node 3)
- **bufferedReader** (Node 3)
- **subjectLine** (Node 4)
- **subjectInfo** (Node 5)
- **subjectName**, **subjectCode**, **fullMarks** (Node 6)
- **subject** (Node 6)
- **line** (Node 7)
- **studentInfo** (Node 8)
- **studentName**, **studentNumber**, **activitiesMark**, **practicalMark**, **midtermExamMark**, **finalExamMark** (Node 9)
- **student** (Node 9)
- **students** (Node 10)
- **subject** (Node 12)

All-USE:

- **fileReader** (Node 3)
- **bufferedReader** (Node 4, 7)
- **subjectLine** (Node 5)
- **subjectInfo** (Node 6)
- **line** (Node 8)
- **studentInfo** (Node 9)
- **students** (Node 12)

ADUP (All-Def-Use Pairs):

- **students** (DEF at Node 1, USE at Node 12)
- **fileReader** (DEF at Node 3, USE at Node 3)
- **bufferedReader** (DEF at Node 3, USE at Nodes 4, 7)
- **subjectLine** (DEF at Node 4, USE at Node 5)
- **subjectInfo** (DEF at Node 5, USE at Node 6)
- **subjectName**, **subjectCode**, **fullMarks** (DEF at Node 6, USE at Node 6)
- **subject** (DEF at Node 6, USE at Node 12)
- **line** (DEF at Node 7, USE at Node 8)
- **studentInfo** (DEF at Node 8, USE at Node 9)
- **studentName**, **studentNumber**, **activitiesMark**, **practicalMark**, **midtermExamMark**, **finalExamMark** (DEF at Node 9, USE at Node 9)
- **student** (DEF at Node 9, USE at Node 10)

Paths through the CFG

1. **Path 1** (Normal execution without exceptions and no students in the file):
    - Start → Node 1 → Node 2 → Node 3 → Node 4 → Node 5 → Node 6 → Node 7 (loop condition false) → Node 12 → Node 13 → End
2. **Path 2** (Normal execution without exceptions and with students in the file):
    - Start → Node 1 → Node 2 → Node 3 → Node 4 → Node 5 → Node 6 → Node 7 (loop condition true)
        - Loop iteration:
            - Node 8 → Node 9 → Node 10 → Node 11 → Node 7 (loop condition false) → Node 12 → Node 13 → End
3. **Path 3** (FileNotFoundException caught):
    - Start → Node 1 → Node 2 → Node 3 (FileNotFoundException) → Node 14 → Node 15 → End
4. **Path 4** (Other exceptions caught):
    - Start → Node 1 → Node 2 → Node 3 → Node 4 → Node 5 → Node 6 → Node 7 (Exception) → Node 16 → Node 17 → End

Detailed Paths with Node Transitions

1. **Path 1** (Normal execution without exceptions and no students in the file):
    - Start → Node 1 (**students = new ArrayList<Student>()**)
    - Node 2 (**try** block starts)

- Node 3 (**fileReader** and **bufferedReader** initialization)
- Node 4 (**subjectLine = bufferedReader.readLine()**)
- Node 5 (**subjectInfo = subjectLine.split(",")**)
- Node 6 (**subject = new Subject(subjectName, subjectCode, fullMarks)**)
- Node 7 (loop condition check, **bufferedReader.readLine() == null**)
- Node 12 (**subject.setEnrolledStudents(students)**)
- Node 13 (**bufferedReader.close()**)
- End

2. **Path 2** (Normal execution without exceptions and with students in the file):
    - Start → Node 1 (**students = new ArrayList<Student>()**)
    - Node 2 (**try** block starts)
    - Node 3 (**fileReader** and **bufferedReader** initialization)
    - Node 4 (**subjectLine = bufferedReader.readLine()**)
    - Node 5 (**subjectInfo = subjectLine.split(",")**)
    - Node 6 (**subject = new Subject(subjectName, subjectCode, fullMarks)**)
    - Node 7 (loop condition check, **bufferedReader.readLine() != null**)
        - Loop iteration:
            - Node 8 (**studentInfo = line.split(",")**)

        - Node 9 (**Student student = new Student(studentName, studentNumber, activitiesMark, practicalMark, midtermExamMark, finalExamMark)**)
        - Node 10 (**students.add(student)**)
        - Node 11 (loop back to Node 7)
    - Node 7 (loop condition check, **bufferedReader.readLine() == null**)
    - Node 12 (**subject.setEnrolledStudents(students)**)
    - Node 13 (**bufferedReader.close()**)
    - End
3. **Path 3** (FileNotFoundException caught):
    - Start → Node 1 (**students = new ArrayList<Student>()**)
    - Node 2 (**try** block starts)
    - Node 3 (FileNotFoundException)
    - Node 14 (**catch(FileNotFoundException e)**)
    - Node 15 (**System.out.print("File was not found!")**)
    - End
4. **Path 4** (Other exceptions caught):
    - Start → Node 1 (**students = new ArrayList<Student>()**)
    - Node 2 (**try** block starts)
    - Node 3 (**fileReader** and **bufferedReader** initialization)
    - Node 4 (**subjectLine = bufferedReader.readLine()**)

- Node 5 (**subjectInfo = subjectLine.split(",")**)
- Node 6 (**subject = new Subject(subjectName, subjectCode, fullMarks)**)
- Node 7 (loop condition check, **bufferedReader.readLine() != null**)
    - Loop iteration:
        - Node 8 (**studentInfo = line.split(",")**)
        - Node 9 (**Student student = new Student(studentName, studentNumber, activitiesMark, practicalMark, midtermExamMark, finalExamMark)**)
        - Node 10 (**students.add(student)**)
        - Node 11 (loop back to Node 7)
- Node 7 (loop condition check, **bufferedReader.readLine() != null**)
- Exception occurs at any point within the try block
- Node 16 (**catch(Exception e)**)
- Node 17 (**System.out.print("Error, Could not process the file!")**)
- End