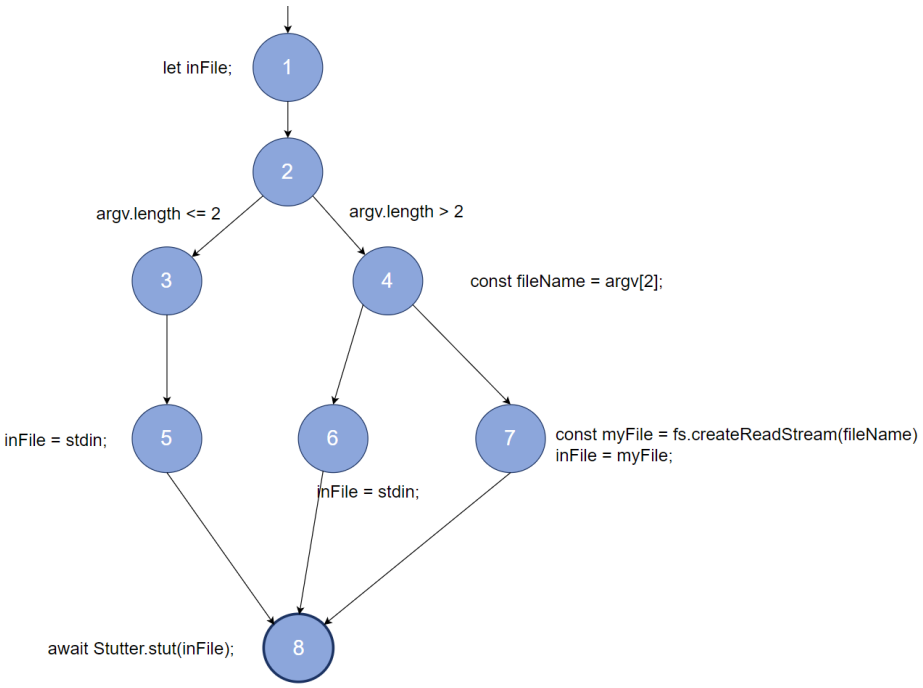
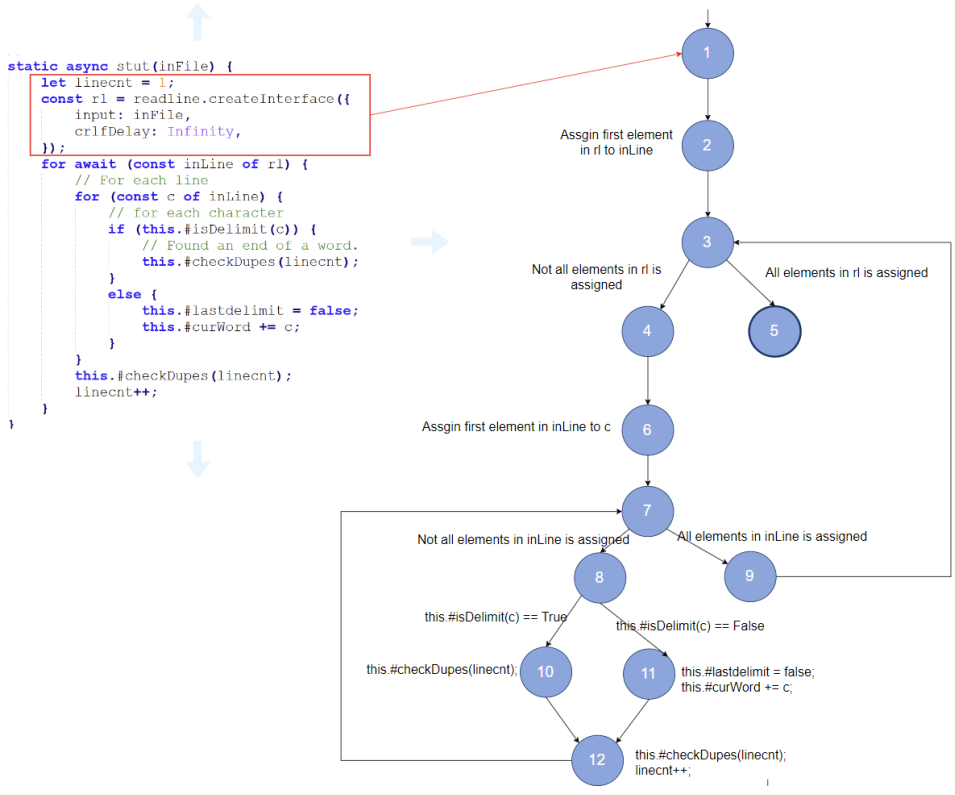
Use "Stutter.js " for questions a-d. (25% x 4)

(a) Draw control flow graphs for all the functions defined in "Stutter.js".

main()

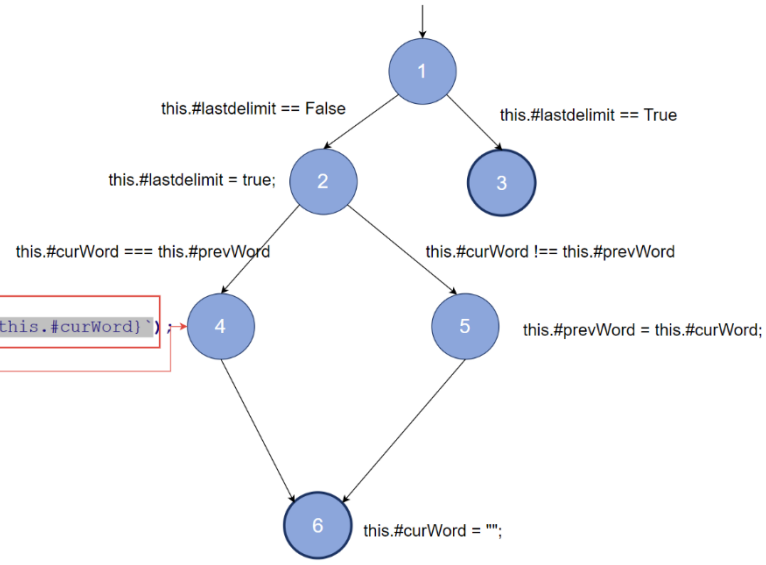


stut()

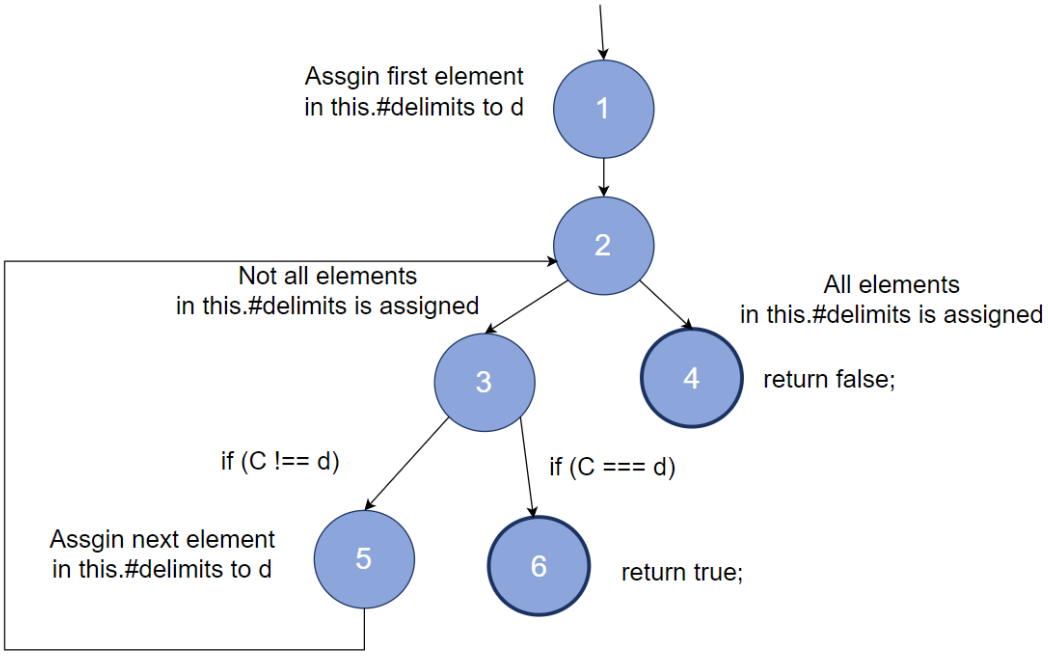


checkDupes()

```
static #checkDupes(line) {
    if (this.#lastdelimit)
        // already checked, keep skipping
        return;
    this.#lastdelimit = true;
    if (this.#curWord === this.#prevWord) {
        console.log(
            `Repeated word on line ${line}: ${this.#prevWord} ${this.#curWord}`);
    }
    else {
        this.#prevWord = this.#curWord;
    }
    this.#curWord = "";
}
```



isDelimit()



(b) List all the call sites of the functions defined in "Stutter.js".

Line	Call site
112	Stutter.stut(inFile);
44	this.#isDelimit(c)
46	this.#checkDupes(linecnt);

53	this.#checkDupes(linecnt);

(c) List all coupling du-pairs for each call site in (b).

	Last-Def	**First-Use**
1	(main(),inFile, 98)	(stut(),inFile, 37)
2	(main(),inFile, 104)	(stut(),inFile, 37)
3	(main(),inFile, 109)	(stut(),inFile, 37)
4	(stut(), c, 42)	(isDelimit(), C, 84)
5	(stut(),linecnt, 35)	(checkDupes(),line, 70)
6	(stut(),linecnt, 54)	(checkDupes(),line, 70)
7	(isDelimit(), , 85)	(stut(), , 44)
8	(isDelimit(), , 86)	(stut(), , 44)

(d) **Create test data to satisfy All-Coupling Uses Coverage for "Stutter.js".**

Informally, to cover all coupling du-pairs in (c).

Case #	**argv.length**	**args[2]**	**inFile**
1	1		“TestData”
2	2	“”	“TestData”
3	2	“Input.txt”	“TestData”
4	2	“Input.txt”	“Test\tData”
5	2	“Input.txt”	“Test Test”
6	2	“Input.txt”	“Test\nTest”
7	2	“Input.txt”	“Test\tData”
8	2	“Input.txt”	“TestData”