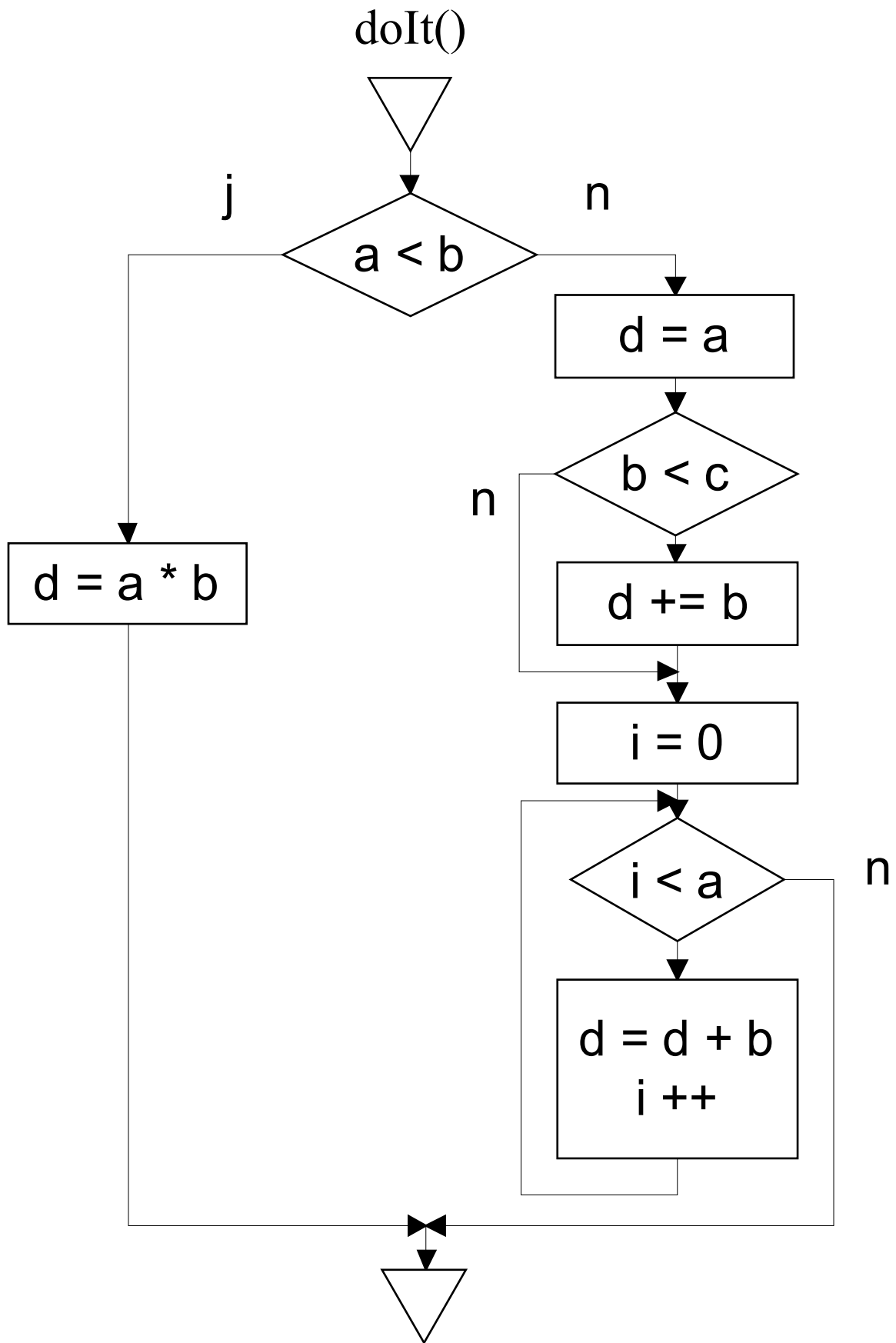


```
int doIt (int a, int b, int c) {  
    int d;  
    if (a < b)  
        d = a * b;  
    else {  
        d = a;  
        if (b < c)  
            d += b;  
        for (int i = 0; i < a; i++) {  
            d = d + b;  
        }  
    }  
    return d;  
}
```



Selbstorganisierende Liste

Eine *selbstorganisierende Liste* ist eine Liste, bei der kurz aufeinanderfolgende Zugriffe auf dieselben Elemente sehr effizient sind: Wird ein Element in der Liste gefunden, entfernt man es aus der Liste und fügt es am Anfang der Liste wieder ein. Bei nachfolgenden Zugriffen auf dieses Element wird es (am Anfang der Liste) sehr schnell gefunden.

Implementieren Sie die Methoden

- **prepend** zum Erzeugen und Einfügen eines neuen Elements am Anfang der Liste,
- **find** zum Suchen und zum Verschieben an den Anfang eines Elements mit gegebenem *key*, wird das Element nicht gefunden, soll *null* zurückgegeben werden und
- **remove** zum Löschen eines Elements mit gegebenem *key*.

Hinweis: Sie können in remove die Methode find verwenden (Sonderfälle beachten!)

```
class SelfOrgList {
    Node head;

    SelfOrgList() { head = null; }
    void prepend(int key, String val) {...}
    String find(int key) {...}
    String remove(int key) {...}
} // SelfOrgList
```

```
class Node {
    Node next;
    int key;
    String val;
    Node(int key, String val) {
        ...
    } // Node
} // Node
```