

3.3 Probleme der automatischen Garbage Collection

Der nachfolgende Klasse Message erzeugt Nachrichten, von denen eine auf der Konsole ausgegeben wird.

```
import java.util.ArrayList;
import java.util.List;

public class Message {
    private static List<message> messageCache = new ArrayList<>();
    private static int count = 0;
    private String msg;

    public Message(String txt){
        count++;
        this.msg = String.format("Msg (##d) %s",count,txt);
    }

    public void putInCache() {
        messageCache.add(this);
    }

    public void writeMessage(){
        System.out.println(this.msg);
    }

    public static Message getMessage(int index) throws Exception {
        if (index < 0 || index > messageCache.size()) {
            throw new Exception("Message is not in cache any more.
Please load it from disk!");
        }

        return messageCache.get(index);
    }

    public static void main(String... args){
        for(int i=0; i<10000000;i++){
            // message received here
            Message msg = new Message("test");
            msg.putInCache();
        }

        try {
            Message.getMessage(42).writeMessage();
        } catch(Exception e) {
            System.out.println(e.toString());
        }
    }
}
```

```
}
```

Welche Probleme im Code verursachen dieses Verhalten und wie würden Sie dies lösen?

- Es wird ein statischer Cache befüllt. Nach dem Ablauf der füllenden for-Schleife existiert ein weiterer lesender Zugriff auf den cache. Für den Java-Compiler ist es nicht möglich zu erkennen, dass nur ein einzelnes Element aus dem sehr großen Cache benötigt wird. Die Referenzen auf alle neu erstellten Nachrichten werden in diesem Cache gehalten und können vor dem Beenden der main- Methode nicht freigegeben werden. Der GC räumt lediglich die lokale Referenz auf die Message Objekte in der for-Schleife auf.
- Caching ist ein Mechanismus, der den Heap über Zeit füllt. Das ist gewünschtes Verhalten. Allerdings sollte es möglich sein, manuell nicht mehr verwendete Messages aus dem Cache zu löschen, oder ihn in seiner Größe zu begrenzen, oder ein automatisches Clean-Up nach Zeit, oder nach Speicher einzuführen.