

Verifica di programmi con openJML

Esercizio 1

COPIA il codice JML dell'esercitazione precedente (ad esempio Account) e prova con ESC i contratti.

Rimuovi System.out e classi di prova con main.

Esercizio 2 (cartella esercizio 5)

Prova alcuni contratti semplici (come quelli visti a lezione)

Esercizio 3 (cartella esercizio 6):

Considera il seguente problema. Ho una classe che ha un campo array di 5 interi tutti inizializzati a 0.

Voglio provare che il costruttore inizializza correttamente il campo array.

Procedo per gradi:

- se metto anche semplicemente il contratto `endures array != null` senza invarianti non riesco a provare nulla
- aggiungo un loop invariant che dice che array non è nullo
- aggiungo la post condizione che array abbia `length == 5` e modifico il loop invariant in accordo
- aggiungo anche la condizione che gli elementi dell'array sono messi a 0 come desiderato
- se aggiungo `normal_behavior`, devo anche dimostrare che l'indice è corretto

Esercizio 4:

Cerca di dimostrare i contratti non chiusi dell'esercizio 1

Esercizio 5 - te giugno 21

Dato il seguente programma in Java, che controlla se un array contiene un numero pari maggiore di b;

```
boolean check(int[] a, int b) {
    if (a != null && a.length > 0) {
        for (int x : a) {
            if (x > b && x % 2 == 0)
                return true;
        }
    }
    return false;
}
```

Scrivi i contratti e prova la correttezza. Puoi semplificare il codice se i contratti che metti rendono superflui alcuni controlli. Puoi riscrivere il codice a tuo piacimento per portare a termine la dimostrazione. Ricordati la sintassi dei quantificatori:

`(\forall <dominio>; <range_valori>; <condizione>)`

`(\exists <dominio>; <range_valori>; <condizione>)`