(Detection && Removal) of {Redundant Casts in Amplification Testing}

1. @Why?

Readability!

```
// DSpot Amplified Code
((int) (((FilterStreamType)
    (type)).hashCode())));

// Normal
type.hashCode();
```

97% of cast are redundant

2. @Objective

Make amplification testing more developer friendly

RQ: *Is a fine-grained cast deleter worthwhile compared to a simple cast deleter in terms of accuracy when simplifying superfluous casts?*

3. @Method

Alg. 1 : All Cast Deleter

```
=> ~~(byte) (short) (int)~~ intVariable;
```

Alg. 2 : Double Cast Deleter

```
=> (byte) ~~(short) (int)~~ intVariable;
```

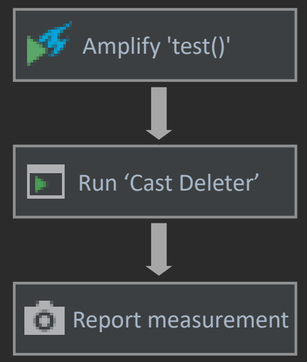
Alg. 3 : IntelliJ's Cast Deleter

› Checks against the hierarchy between the casted and original object.

```
=> ~~(double)~~ (byte) ~~(int)~~ intVariable;
```

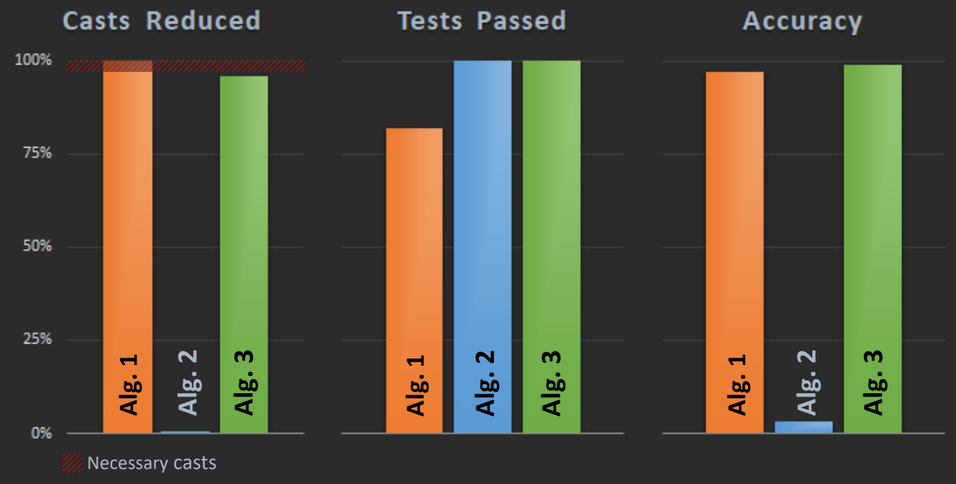
4. @Data Generation

`Repeat for 4 public repositories`



5. @Results

Over 3085 casts in 281 tests

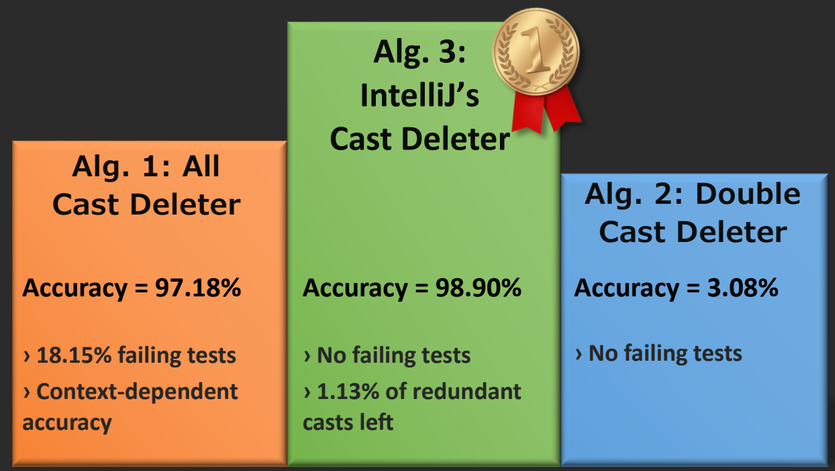


Manual code inspection

- 100% of **necessary casts** are needed for compilation.
- 89.66% of **necessary casts** are needed in declaration statements.

  `E.g. Child o = (Child) parentType;`

6. @Conclusion



/** *@author* George Vegelien *on* 30-06-2021 *@email* g.j.b.vegelien@student.tudelft.nl *@supervisor* Carolin Brandt *@professor* Andy Zaidman */

CSE3000

TU Delft