

Static Analysis Handin #6
 Alexander B. Hansen, 20062813
 Sivert Bertelsen, 20062179
 Søren Løbner, 20050677
 {alx,sivert,lobner}@cs.au.dk

Følgende program analyseres

```
q = malloc;
y = &q;
x = *y;
p = null;
y = &p;
*x = p;
```

Andersen analyse, constrains:

```
{malloc} ⊆ [[q]]
{&q} ⊆ [[y]]
&id ∈ [[y]] ⇒ [[id]] ⊆ [[x]]
{&p} ⊆ [[y]]
&id ∈ [[x]] ⇒ [[p]] ⊆ [[id]]
```

Løsningen til disse constrains er:

```
[[q]] = {malloc}
[[y]] = {&q, &p}
[[x]] = {malloc, null}
[[p]] = {null}
```

Shape analyse, constrains:

```
[[q = malloc]] = JOIN(q = malloc)↗q u {(&q, malloc)}
[[y = &q]] = JOIN(y=&q)↗y u {(&y, &q)}
[[x = *y]] = right(JOIN(x=*y), x, y)
[[p = null]] = JOIN(p=null)↗p
[[y = &p]] = JOIN(y=&p)↗y u {(&y, &p)}
[[*x = p]] = left(JOIN(*x=p), x, p)
```

Løsning til constrains.

```
[[q = malloc]] = ∅ u {(&q, malloc)}
[[y = &q]] = {(&q, malloc)} u {(&y, &q)} = {(&q, malloc), (&y, &q)}
[[x = *y]] = right({(&q, malloc), (&y, &q)}, x, y)
           = {(&q, malloc), (&y, &q)} u {(&x, malloc)}
           = {(&q, malloc), (&y, &q), (&x, malloc)}
[[p = null]] = {(&q, malloc), (&y, &q), (&x, malloc)}
[[y = &p]] = {(&q, malloc), (&y, &q), (&x, malloc)}↗y u {(&y, &p)}
           = {(&q, malloc), (&x, malloc), (&y, &p)}
[[*x = p]] = left({(&q, malloc), (&x, malloc), (&y, &p)}, x, p)
           = {(&q, malloc), (&x, malloc), (&y, &p)} u ∅
```

Dvs i sidste programpunkt vil

```
pt(q) = {malloc}
pt(x) = {malloc}
pt(y) = {&p}
```

$pt(p) = \{null\}$

I dette program punkt P haves i programpunkt v for variabel x at
 $nullable(Shape(P,v),v)(x) \neq null \wedge nullable(Andersen(P),v)(x) = ?$

Da Andersen giver at x kan pege på enten malloc eller null, men shape analysen giver at x kun kan pege på malloc.

Dvs at programmet er boostable.