

# Sinteza popravki programa na osnovu ispravnih primera

Ivan Ristović, Milana Kovačević, Strahinja Stanojević

septembar 2018.

# Zadatak

- ▶ Ulaz: Dva isečka koda, jedan koji služi kao specifikacija i drugi sa greškama
- ▶ Zadatak: ispraviti kod da postane semantički ekvivalentan ispravom kodu
- ▶ Ispravljen kod treba da zadrži originalnu strukturu sto je više moguće

# Primeri ulaza

```
int foo1()  
{  
    int x = 1, a = 2;  
    return x + a;  
}
```

```
int foo2()  
{  
    int x = 0, a = 2;  
    return x + a;  
}
```

# Primeri ulaza

## ► Ekvivalentno?

```
int fooEq1()  
{  
    int x = 1;  
    int a = 2;  
    return x + a + 1;  
}
```

```
int fooEq2()  
{  
    return 4;  
}
```

# Primeri ulaza

## ► Sporedni efekti

```
int fooSideEff()  
{  
    print("Hello!");  
    return 1;  
}
```

```
int foo()  
{  
    return 1;  
}
```

# Primeri ulaza

- Moguće evaluirati u nekim slučajevima

```
int anotherFooEq1()  
{  
    int x = 0;  
    x += 3;  
    return x;  
}
```

```
int anotherFooEq2()  
{  
    int a = 0;  
    a++;  
    a++;  
    ++a;  
    return a;  
}
```

# Primeri ulaza

```
int ambFoo1(int x)
{
    if (x >= 0)
        return 1;
    else
        return 0;
}
```

```
int ambFoo2(int y)
{
    return y % 2;
}
```

```
void wrapper1()
{
    int x = ambFoo1(1);
}
```

```
void wrapper2()
{
    int x = ambFoo2(1);
}
```

# Zaključak

► Foo



# Pitanja

???

Hvala na pažnji!