

# Programozási nyelvek és paradigmák

Funkcionális programozás

Kozsik Tamás (2020)

# Művelettel való paraméterezés

- ▶ OOP megközelítés: deferred feature, részleges implementáció
- ▶ FP megközelítés: magasabbrendű függvény

## Eiffel magasabbrendű függvényei: *agent*ekkel

- ▶ call agent (parciális alkalmazás)
- ▶ inline-agent (lambda)

## Call agent

```
agent r                -- agent Current.r
```

```
agent v.r
```

```
agent r(12,24)
```

```
agent v.r(12,24)
```

```
agent v.r(?,24)
```

```
agent {MYCLASS}.r(?,?)
```

```
PROCEDURE[ MYCLASS, TUPLE[INTEGER,INTEGER] ]
```

```
FUNCTION[ MYCLASS, TUPLE[INTEGER,INTEGER], REAL ]
```

```
ROUTINE[ MYCLASS, TUPLE[INTEGER,INTEGER] ]
```

# PROCEDURE

```
feature
  foreach( p: PROCEDURE[ANY,TUPLE[REAL]]; a: ARRAY[REAL] )
  do
    across a as i loop
      p.call([a.item(i)])
    end
  end
```

## Inline-agent

```
agent( x, y: INTEGER ): BOOLEAN do Result := x-y > 5 end
```

- ▶ Lehet elő- és utófeltétele, lokális változói
- ▶ Használhatja a befoglaló osztály feature-jeit
- ▶ Nem használhatja a befoglaló rutin (vagy agent) lokális változóit és formális paramétereit
  - ▶ explicit átadhatók ezek paraméterként