

Zadanie: Indywidualne testowanie SPA przez poszczególnych studentów w Zespole – Iteracja 2

Imie/nazwisko Studenta: Aleksander Anchimiuk

Tablica podsumowująca testy pojedynczych relacji:

| Relacja | Liczba testów |
|----------|---------------|
| Follows | 1 |
| Follows* | 1 |
| Parent | 1 |
| Calls | 2 |
| Modifies | 2 |
| Uses | 1 |

Tablica podsumowująca testy dwóch lub więcej relacji:

| Relacje | Liczba testów |
|--------------------------|---------------|
| Follows, Parent | 1 |
| Follows*, Next, Modifies | 1 |
| Calls, Modifies | 2 |
| Uses, Modifies | 2 |
| Calls, Uses, Modifies | 1 |
| Affects, Next* | 1 |

Łącznie 16 testów

Program źródłowy SIMPLE (dla ułatwienia opisałem nr statementów zgodnie z tym jak nasz program to parsuje):

```
procedure Main {
  z = 30;           // stmt 1
  call compute;    // stmt 2
}
procedure compute {
  a = x + y;        // stmt 3
  b = a - 5;        // stmt 4
  c = b * z;        // stmt 5
  i = 0;            // stmt 6
  while i {         // stmt 7
    x = x + 1;      // stmt 8
    y = y - 1;      // stmt 9
    i = i - 1;      // stmt 10
    if x then {     // stmt 11
      x = z + x;    // stmt 12
    } else {
      y = z - y;    // stmt 13
    }
    call calculate; // stmt 14
  }
  b = z - a + b;    // stmt 15
}
procedure calculate {
  p = x * y + z;    // stmt 16
  q = p - x;        // stmt 17
  r = q + y;        // stmt 18
  x = r * z;        // stmt 19
}
procedure display {
  i = 10;           // stmt 20
  while i {         // stmt 21
    call output;    // stmt 22
    i = i - 1;      // stmt 23
  }
}
procedure output {
  a = x + y + z;    // stmt 24
}
procedure mathHelper {
  a = x + 5;        // stmt 25
  b = y * 2;        // stmt 26
  c = a + b;        // stmt 27
}
```

Testy pojedynczych relacji:

Modifies:

stmt s;

Select s such that Modifies(s, "z")

Oczekiwane wyjście: 1

stmt s;

Select s such that Modifies(s, "x")

Oczekiwane wyjście: 8,12, 19

Calls:

procedure p1;

Select p1 such that Calls(p1, "compute")

Oczekiwane wyjście: Main

procedure p1;

Select p1 such that Calls(p1, "mathHelper")

Oczekiwane wyjście: brak/null

Follows:

Select BOOLEAN such that Follows(3, 15)

Oczekiwane wyjście: FALSE

Follows*:

Select BOOLEAN such that Follows*(3, 15)

Oczekiwane wyjście: TRUE

Parent:

Select BOOLEAN such that Parent(1, 2)

Oczekiwane wyjście: FALSE

Uses:

stmt s;

Select s such that Uses(s, "a")

Oczekiwane wyjście: 4, 15, 27

Testy dwóch i więcej relacji:

Follows, Parent:

stmt s;

Select s such that Follows(6,7) and Parent(7,8)

Oczekiwane wyjście: 8

Calls, Modifies:

procedure p;

Select p such that Calls(p,"compute") and Modifies(p,"z")

Oczekiwane wyjście: Main

procedure p;

Select p such that Calls("Main",p) and Modifies(p,"a")

Oczekiwane wyjście: compute

Uses, Modifies:

Select BOOLEAN such that Uses(11, "x") and Modifies(12, "x")

Oczekiwane wyjście: TRUE

stmt s;

Select s such that Uses(s,"z") and Modifies(s,"x")

Oczekiwane wyjście: 12, 19

Affects, Next*:

Select BOOLEAN such that Next*(8, 12) and Affects(8, 12)

Oczekiwane wyjście: TRUE

Calls*, Uses, Modifies

procedure p;

Select p such that Calls*(p,"calculate") and Uses(p,"y") and Modifies(p,"x")

Oczekiwane wyjście: compute, Main

Follows*, Next, Modifies

procedure p;

Select p such that Follows*(3,15) and Next(8,9) and Modifies(p,"i")

Oczekiwane wyjście: compute