.1 קיימת פרוצדורה Main בקוד והיא יחידה.

```
proc goo() {

proc foo() {

var c2 : char;

var a1, b : string[1000];

var x : int;
}

proc foo() {

proc foo() {

var c2 : char;

var a1, b : string[1000];

var x : int;
}

proc foo() {

var c2 : char;

var a1, b : string[1000];

var x : int;
}

BODY

PROC

ARGS NONE

BODY

PROC

foo

ARGS NONE

BODY

BODY

PROC

2

BODY

BODY

BODY

C2 : char DEC

a1 NEXTP b : string [ATYPE 1000] DEC

X : int
```

לא מקבל ארגומנטים Main .2

3. לא קיימות שתי פונקציות או פרוצדורות עם אותו שם באותו 3

```
proc fee() {
    var a : int;
    }
}

proc fee() {
    var a : int;
    }
}

proc fee() {
    var a : int;
    var a; int;
    }
}

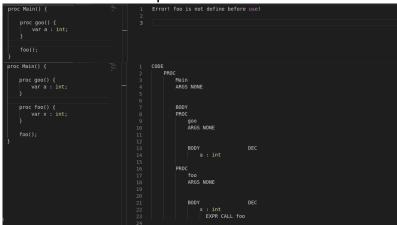
proc fee() {
    var a : int;
    var a; int;
    }
}

proc fee() {
    var a : int;
    var a; int;
    var a; int;
    }
}

proc fee() {
    var a : int;
    var a; in
```

4. לא קיימים שני משתנים עם אותו שם באותו scope

5. פונקציות ופרוצדורות הוגדרו לפני שמפעילים אותן.



6. משתנים הוגדרו לפני שמשתמשים בהם.

```
proc Main() {
    proc goo() {
        a=5;
    }
    proc foo() {
        var x : int;
    }
}

proc foo() {
    var a : int;
    a=5;
    }

proc foo() {
    var a : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
    var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    }

proc foo() {
        var x : int;
    a=5;
    a=5
```

7. כמות הארגומנטים בקריאה לפונקציה/פרוצדורה צריכה להיות שווה לכמות הארגומנטים בהגדרת הפונקציה/פרוצדורה.

```
proc goo() {
    var a : int;
    a=5;
    }

proc Main() {
    proc goo() {
    var a : int;
    }

proc do(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

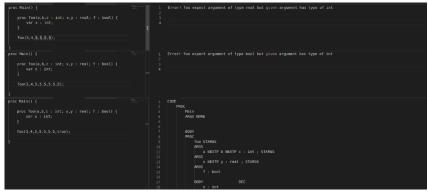
proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    }

proc foo(a,b,c : int) {
    var a : int;
    a=5;
    a=
```

8. טיפוסים של הארגומנטים בקריאה לפונקציה/פרוצדורה תואמים לטיפוסים בהגדרת הפונקציה/פרוצדורה.



 טיפוס הערך המוחזר מהפונקציה תואם לטיפוס ההחזרה המוכרז בכותרת של הפונקציה. וטיפוס ההחזרה של הפונקציה לא יכול להיות מחרוזת.

10. טיפוס הערך המוחזר מהפונקציה תואם לטיפוס המשתנה שלתוכו נכנס הערך שמוחזר מהפונקציה.

```
proc Main() {
    func foo(a : int) return int {
        return 0;
    }

    var a : char;
    a = foo(3);
}

proc Main() {
    func foo(a : int) return int {
        return 0;
    }

proc Main() {
    func foo(a : int) return int {
        return 0;
    }

    var a : int;
    a = foo(3);
}

var a : int;
    a = foo(3);

}

return 0;

re
```

11. טיפוס התנאי ב if -הוא מטיפוס

bool וב for וב- while מטיפוס.12

```
Error! The Condition in 'IF' statment must be of boolean type
    var x : int;
                                                                               BODY DEC
x : int IF
COND > ATYPE USEID x ATYPE 5
                                                                      Error! The Condition in 'WHILE' statment must be of boolean type
proc Main() {
    while (x*8) {
    var z: int;
    z=3;
                                                                     CODE
PROC
Main
ARGS NONE
                                                                                                 DEC
WHILE COND ATYPE true bool
                                                                                                 EXPR = ATYPE USEID z ATYPE 3
                                                                      CODE
PROC
proc Main() {
                                                                                Main
ARGS NONE
                                                                                     int = i COND < ATYPE USEID i ATYPE 3 + ATYPE USEID i ATYP
                                                                                                     EXPR = ATYPE USEID w ATYPE 3
```

- .int טיפוס הביטוי המופיע כאינדקס ב-[] של מחרוזת הוא מטיפוס
 - .14 לא משתמשים באופרטור [] בשום טיפוס חוץ ממחרוזת.

```
proc Main() {
    var x : char*;
    var y['d']: string;
}

proc Main() {
    var x[0] : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

proc Main() {
    var x : char*;
    var y[5]: string;
}

BODY

BODY

BODY

BODY

DEC

x : char* DEC

y : string[] [ ATYPE 5 int ]
```

15. טיפוס של המשתנה מצד שמאל של האופרטור השמה (=) תואם לטיפוס הביטוי מצד ימין.

```
proc Main() {
    var a : int;
    a = 5.5;
}

proc Main() {
    var b : char;
    b = 5;

proc Main() {
    var c: real;
    c = 'd';
}

proc Main() {
    var d: string;
    d = true;
}

proc Main() {
    var e: bool;
    e = 'hhfgh";
}

proc Main() {
    var a: int;
    var b: char;
    var b: string;
    d = true;
}

proc Main() {
    var a: int;
    var b: char;
    var b: char;
    var b: char;
    var b: char;
    var c: real;
    c = 'shfgh";
    var b: char;
    var c: real;
    var c: real;
    var c: string;
    d = true;
}

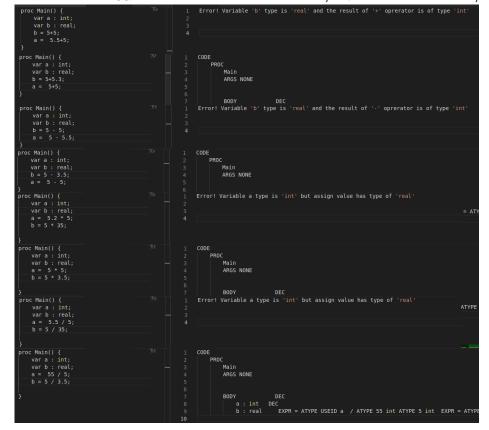
proc Main() {
    var a: int;
    var b: char;
    var b: char;
    var b: char;
    var c: real;
    var c: real;
    var c: real;
    var c: real;
    var b: char;
    var b: char;
    var c: real;
    var d: string;
    d = true;
}

proc Main() {
    var a: int;
    var b: char;
    var b: char;
    var c: real;
    var d: string;
    var d: string be' define type is 'thar' but assign value has type of 'thar'

    var d: string be' define type is 'thar' but assign value has type of 'thar'

    var d: string be' define type is 'thar' but assign
```

16. עבור אופרטורים (+,-,*,-) שני האופרנדים חייבים להיות int או real. והתוצאה היא , int אם שני האופרנדים , int אחרת הטיפוס של הביטוי הוא , int



.bool שני האופרנדים חייבים להיות שני ($\|.\&.$) שני האופרנדים חייבים להיות שני אופרטורים ($\|.\&.$)

```
proc Main() {
    var a : bool;
    var b : bool;
    a = 55 && 5;
}

proc Main() {
    var a : bool;
    var b : bool;
    var b : bool;
    a = true && b;
}

proc Main() {
    var a : bool;
    var b : bool;
    var b : bool;
    a = true || 'h';
}

proc Main() {
    var a : bool;
    var b : bool;
    a = true || 'h';
}

proc Main() {
    var a : bool;
    var b : bool;
    a = true || false;
}

proc Main() {
    var a : bool;
    var b : bool;
    a = true || false;
}

proc Main() {
    var a : bool;
    var b : bool;
    a = true || false;
}

BODY

BODY

BODY

BODY

DEC

a : bool EXPR = ATYPE USEID a || ATYPE true bool ATYPE

BODY DEC

b : bool EXPR = ATYPE USEID a || ATYPE true bool ATYPE
```

והתוצאה real או int שני חייבים חייבים להיות שני (<,>,<,>) שני האופרנדים חייבים להיות bool התוצאה bool היא



19. עבור האופרטורים (==,!=) שני האופרנדים יכולים להיות שני int/real/bool/char/pointers

```
Error! '!=' operator is available for same types only!
proc Main() {
     var b : bool;
var c : real;
b = a != a;
                                                                      CODE
    var a : int;
var b : bool;
var c : real;
b = 's' != 's';
                                                                            PROC
                                                                                  Main
                                                                                  ARGS NONE
                                                                                  BODY
                                                                                       a: int DEC
b: bool DEC
                                                                                                         EXPR = ATYPE USEID b != ATYPE 's' char
                                                                       Error! '==' operator is available only for same types
proc Main() {
    var a : int;

var b : bool;

var c : real;

b = "sddsd" == 3;

b = 5.20 == 1.2;
proc Main() {
   var a : int*;
   var b : bool;
                                                                             PR0C
                                                                                  Main
     var c : int*;
b = 50 == 30;
                                                                                  ARGS NONE
     b = 5.20 == 1.2;
                                                                                  BODY
                                                                                                        DEC
                                                                                                          EXPR = ATYPE USEID b == ATYPE 50 int AT
```

.int יכול להיות מופעל על מחרוזות והתוצאה היא 20. אופרטור ערך מוחלט (| |) יכול להיות מופעל על

```
proc Main() {
    var a : string;
    var b: int;
    b = |'d'|;

}

proc Main() {
    var a : string;
    var b: int;
    b = |a|;
    b = |a|;

}

CODE

PROC

PROC

ARGS NONE

BODY

B
```

21. אופרטור (!) יכול להיות מופעל על bool והתוצאה היא bool.

```
proc Main() {
    var a : bool;
    var b : int;
    a = !b;
}

proc Main() {
    var a : bool;
    var b : int;
    a = !true;

}

CODE

PROC

Main

ARGS NONE

BODY

BODY
```

תוצאה למצביע int/real/char/string פופעל רק על משתנים מטיפוס & מופעל רק על משתנים.22

```
proc Main() {
    var a : int;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : bool;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var a : int*;
    var a : int*;
    var b : real;
    a = &b;
}

proc Main() {
    var a : int*;
    var a : int
```

23. אופרטור ^ מופעל רק על מצביעים

```
proc Main() {
    var a: int*;
    var b: bool;
    a = ^b;
}

proc Main() {
    var a: int;
    var b: bool;
    a = ^b;
}

proc Main() {
    var a: int;
    var b: bool;
    a = ^b;
}

proc Main() {
    var a: int;
    var b: bool;
    a = ^b;
}

proc Main() {
    var a: int;
    var b: real*;
    a = ^b;
}

Body

Body
```