Patito++ User Manual

Video Demo: ***https://youtu.be/yI-PUdtwmiM***

Structure of a program in Patito++:

```
program nameOfProgram;

main() {

}
```

Declaring a Variable (int, float, char) and assigning value :

```
program nameOfProgram;
var int i;

main() {
    var float a, b;
    a = 0;
    b = 0;
}
```

Declaring an Array or Matrix (int, float, char) and assigning value :

```
program arrayfind;
var int array1[3];

main() {
    var int matrix1[2][2];

    array1[0] = 2;
    array1[1] = 8;
    array1[2] = 5;

    matrix1[0][0] = 1;
```

```
    matrix1[1][0] = 3;
    matrix1[1][1] = 2;
    matrix1[0][1] = 6;

    print(array1[4]);
    print(matrix1[1][1]);
}
```

Declaring and calling a function:

Return statement must be placed on any function which is non void and all void functions must not have return statements.

```
program myprogram;

function int add(int a, int b) {
    return(a + b);
}

main() {

   var int i;
   i = add(10, 15);

    print(i);
}
```

Reading and printing an input value from the console:

```
program myprogram;

main() {
   var int i;

   read( i );

   print( i );
}
```

Using conditional statements and boolean operators:

Less than → <
Greater than → >
Equals → ==
Not equals → <>
And → &
Or → |

```
program fact;

main() {
    var int c;
    c = 5;
    if (c > 1) then {
        print(“Yes”);
    } else {
        print(“No”);
    }
}
```

Using WHILE and FOR loops:

```
program myprogram;

main() {
    int i = 0;
    while (i < 5) {
        print( i )
        i = i + 1;
    }

    for i = 0 to i < 5 {
        print( i );
    }
}
```

Using Matrix Operators:

Determinant → $

```
program myprogram;

main() {
    var float result;
        int i, j, matrix[3][3];

    %% assign matrix
    matrix[0][0] = 2;
    matrix[1][0] = 2;
    matrix[2][0] = 1;
    matrix[0][1] = 0 - 3;
    matrix[1][1] = 0;
    matrix[2][1] = 4;
    matrix[0][2] = 1;
    matrix[1][2] = 0 - 1;
    matrix[2][2] = 5;

    result = matrix$;

    print(result);
}
```

Transpose → !

```
program matrixtranspose;

main() {
    var int i, j, matrix[2][3], result[3][2];

    %% assign matrix
    matrix[0][0] = 1;
    matrix[1][0] = 2;
    matrix[0][1] = 3;
    matrix[1][1] = 4;
    matrix[0][2] = 5;
    matrix[1][2] = 6;

    print("Matrix assigned:");
    for j = 0 to j < 3 {
        for i = 0 to i < 2 {
            print(matrix[i][j]);
        }
    }
```

```
    result = matrix!;

    print("Result matrix:");
    for j = 0 to j < 2 {
        for i = 0 to i < 3 {
            print(result[i][j]);
        }
    }
}
```

Inverse → ?

```
program matrixinverse;

main() {
    var int i, j, matrix[3][3];
        float result[3][3];

    %% assign matrix
    matrix[0][0] = 0 - 1;
    matrix[1][0] = 2;
    matrix[2][0] = 3;
    matrix[0][1] = 0 - 2;
    matrix[1][1] = 1;
    matrix[2][1] = 4;
    matrix[0][2] = 2;
    matrix[1][2] = 1;
    matrix[2][2] = 5;

    result = matrix?;

    print("Result matrix:");
    for j = 0 to j < 3 {
        for i = 0 to i < 3 {
            print(result[i][j]);
        }
    }
}
```