

## Testing

The main testing strategy used was a series of automated unit testing functions, which create an automatic test report in a file called 'test\_results.txt'.

These functions and their prototypes are all detailed in 'testing.c'.

Testing mode is activated by running the program with 'testing' as argv[2] and will generate a new 'test\_results.txt' file each time it is run.

If tests are successful there will also be the message 'couldn't open file, madeup.txt' printed to the console. This is not an error.

These unit tests are designed to test each individual function against a number of test cases to check that data is being correctly assigned and passed and that functions are behaving as intended.

Any functions that included an error message and exit() command were intended to be tested using White Box testing although in practice only the 'Varnum', 'Var' and 'Number' functions are tested this way.

## White Box Testing

Print statements during testing as well as test cases that rely on function getting to correct place in functions

## Black Box Testing

Automated tests for any functions setting variables or returning variables to ensure they do the correct thing.

Some tests load dummy files for testing, others use testing versions of structures and variables to ensure correct functionality.

## Bugs Found During Testing

Many. But some examples found while using automated testing are as follows:

- Polish grammar was allowed to parse phrases in the form SET A := \* ;
- Set draw function was setting both x and y co-ordinates to the same value
- Set draw function wasn't rounding correctly
- Stack top value for Polish function was not being set correctly
- The program counter was not moving on correctly in a number of functions

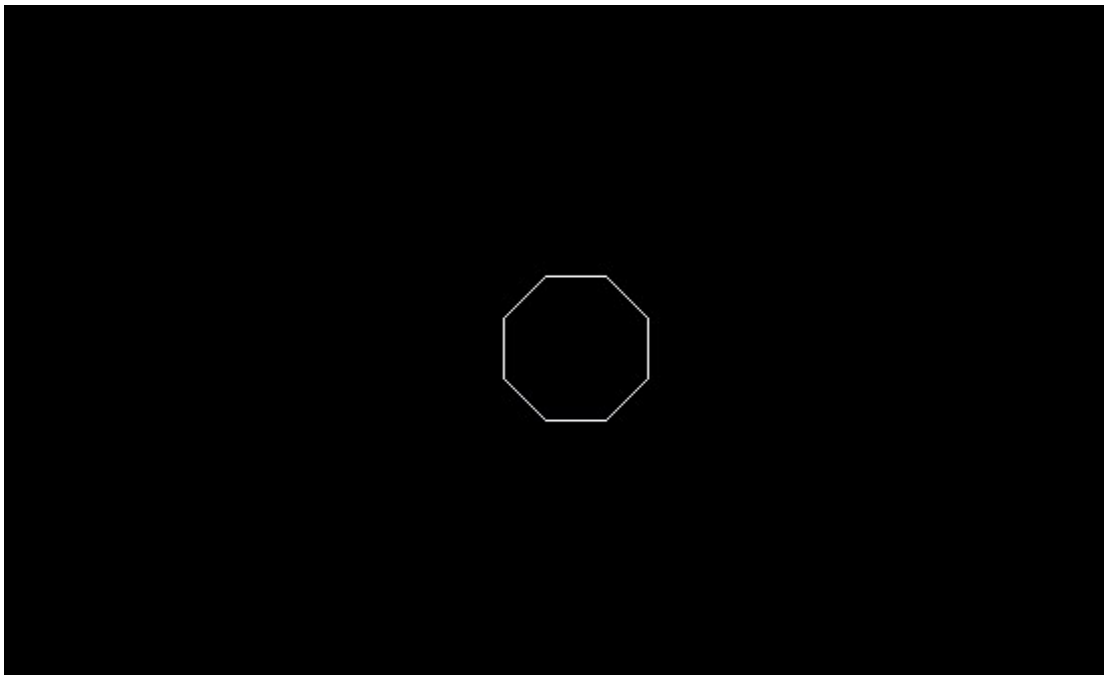
### Other Testing

Tested with a number of turtle files to ensure output in SDL was correct according to grammar specified.

Also tested incorrect grammar to ensure error messages are displayed correctly and the program is exited safely.

```
{  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
    FD 30  
    LT 45  
}
```

Produces the following:

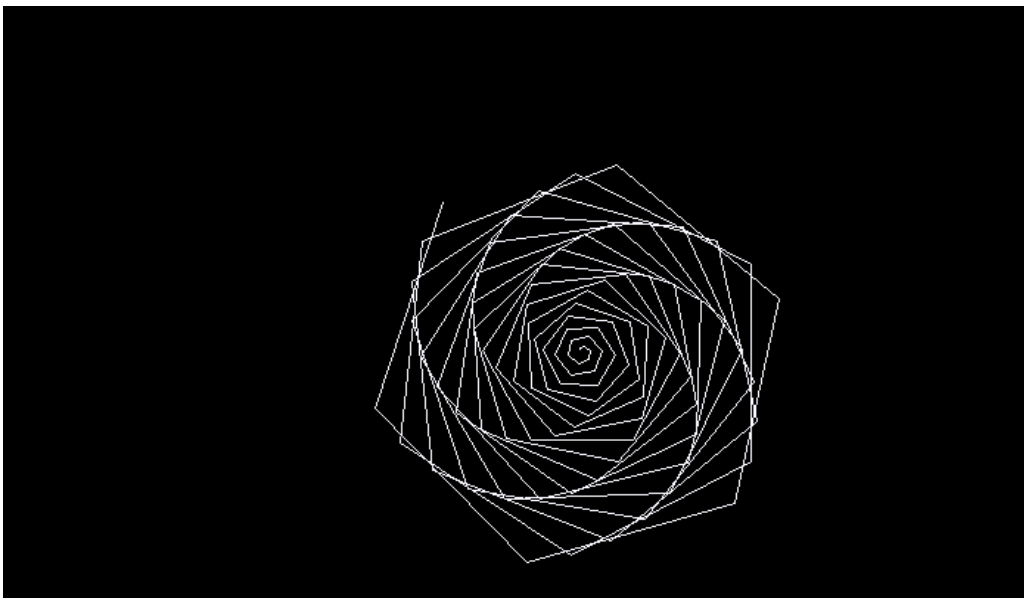


```

{
    DO A FROM 1 TO 100 {
        SET C := A 1.5 * ;
        FD C
        RT 62
    }
}

```

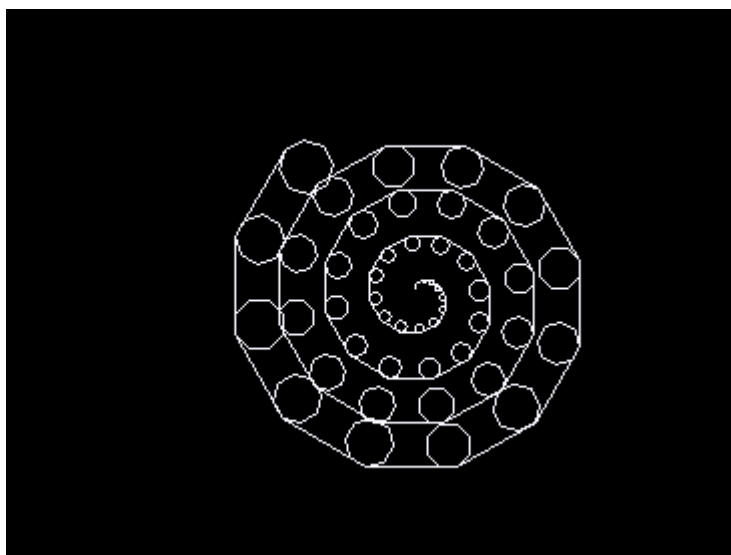
Produces the following:



```

{
    DO A FROM 1 TO 50 {
        FD A
        RT 30
        DO B FROM 1 TO 8 {
            SET C := A 5 / ;
            FD C
            RT 45
        }
    }
}

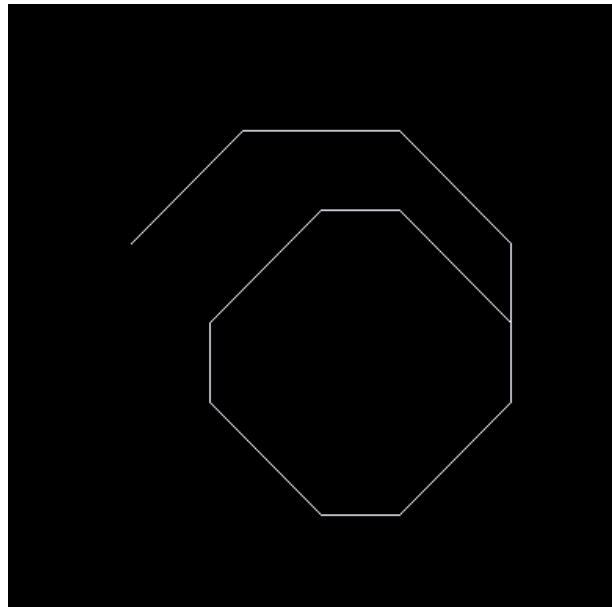
```



```

{
    DO A FROM 1 TO 8 {
        RT 45
        FD 100
        IF ( A > 4 ) {
            RT 45
            FD 50
        }
    }
}

```



```

{
    DO A FROM 1 TO {
        SET C := A 1.5 * ;
        FD C
        RT 62
    }
}

```

```

Invalid VAR? It happened on line 2 in turtle_test3.txt
The current word is {
previous word = TO

```