

THE PEER LEAD TEAM LEARNING EXPERIENCE

(PLTL)

March 28 – 30, 2019

MAA Oklahoma-Arkansas Regional Conference
Northeastern State university in Tahlequah, Oklahoma
Presenter: Bradford House

WHAT IS MY ROLE?

These roles include:

Learning Julia and its many mathematics function so that I can use as well as teach others how to use it.

Writing user friendly functions in the Julia code based on the syllabi of the various courses utilizing the PLTL workshops.

Teaching students how to code in Julia as well as how to break down functions in mathematics so that they can be written and solved in Julia code



WHAT IS JULIABOX?

LEARNING JULIA

In my opinion, Julia is a very good solution when looking for a low-cost, easy to use but efficient way to show students how to create simple code or write a simple program.

This would be because:

It is not necessary for users to download an additional compiler to because the website doubles as a functional compiler for the Julia language.

This also makes it inexpensive for students because there is no required cost for this resource and no additional resources that must be downloaded making it accessible from any computer.



Julia is also very easy to interpret as well as memorize and so it is very easy to teach it to people with very little background with writing code or programs making it perfect for the students experiencing the workshop.



Despite how free flowing and easy to use Julia is it is also diverse in the number of built-in functions that can be used in the language and these functions are simple making it perfect for the Peer Lead Team Learning Experience as the students being introduced to it are novices.

LEARNING JULIA

Some of these functions include:

- `sin()`
- `cos()`
- `arctan()`
- `plot()` (used for graphing)

LEARNING JULIA

Solve Quadratics

```
In [ ]: function quadratic(a,b,c)
    root1 = (-b+sqrt((b^2)-(4*(a)*(c))))/(2*a)
    root2 = (-b-sqrt((b^2)-(4*(a)*(c))))/(2*a)
    println("The roots are", root1, " and ", root2)
end

println("Enter the coefficient of x squared")
a = parse{Int64,readline()}
println("Enter the coefficient of x")
b = parse{Int64,readline()}
println("Enter the last term")
c = parse{Int64,readline()}

quadratic(a,b,c)
```

WRITING USER
FRIENDLY
FUNCTIONS
SOLVING
QUADRATICS

Midpoint Formula

```
In [ ]: function midpoint(x1,x2,y1,y2)
        xco = (x1+x2)/2
        yco = (y1+y2)/2

        println("The midpoint is: (",xco,",",yco,")")
    end

    println("Enter first x value")
    w = parse{Int64,readline()}
    println("Enter second x value")
    x=parse{Int64,readline()}
    println("Enter first y value")
    y=parse{Int64,readline()}
    println("Enter second y value")
    z=parse{Int64,readline()}

    midpoint(w,x,y,z)
```

Finding Slope Intercept

```
In [ ]: println("Enter x value")
        x = parse{Int64,readline()}
```

MIDPOINT FORMULA

Distance Formula

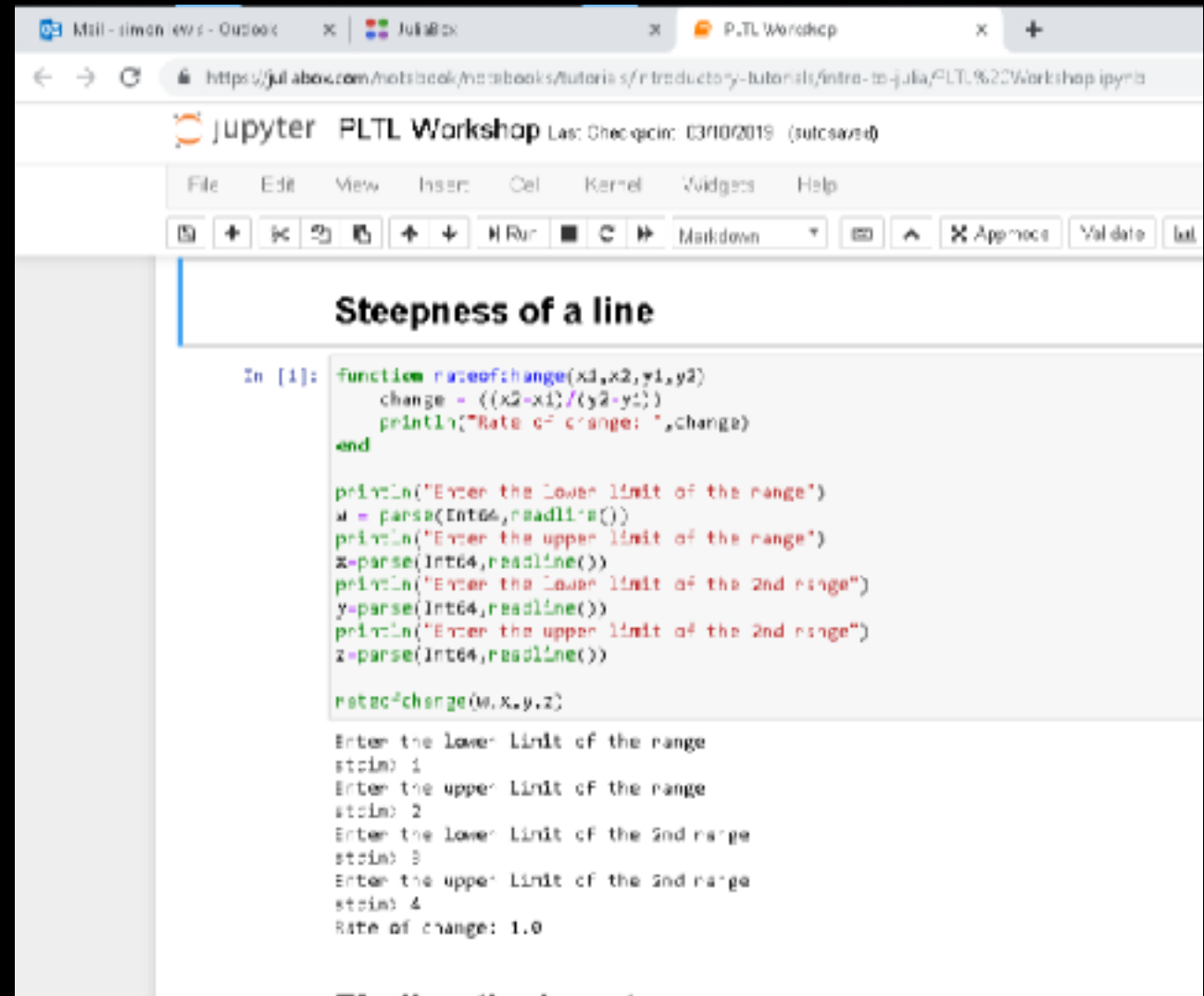
```
In [ ]: function distance(x1,x2,y1,y2)
        length = sqrt(((x1-x2)^2)+((y1-y2)^2))
        println("The distance is: ", length)
        end

        println("Enter first x value")
        w = parse{Int64,readline()}
        println("Enter second x value")
        x=parse{Int64,readline()}
        println("Enter first y value")
        y=parse{Int64,readline()}
        println("Enter second y value")
        z=parse{Int64,readline()}

        distance(w,x,y,z)
```

DISTANCE
FORMULA

STEEPNESS OF A LINE

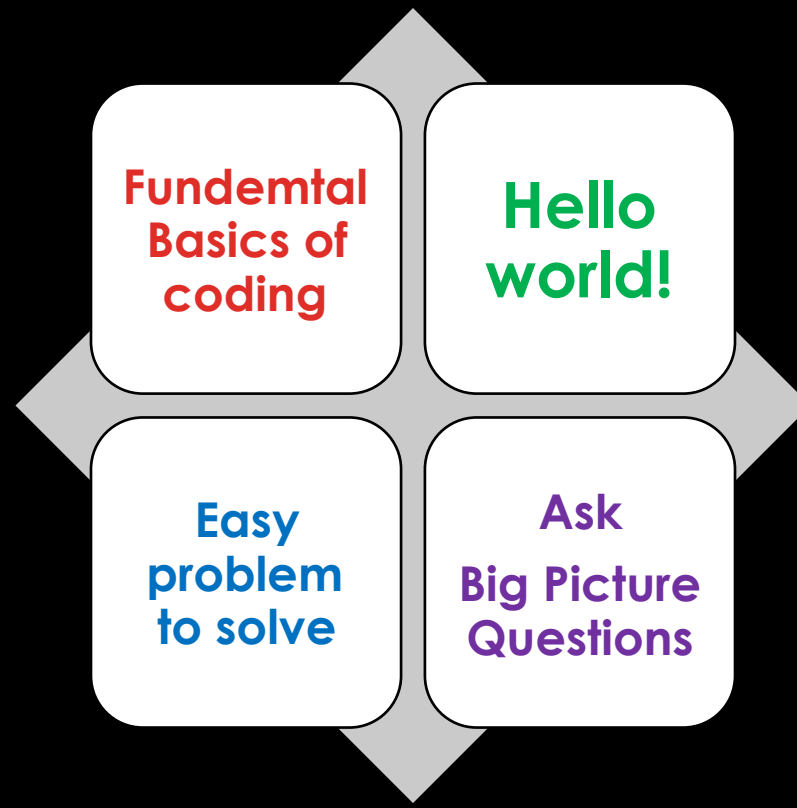


```
In [1]: function rateofchange(x1,x2,y1,y2)
        change = ((x2-x1)/(y2-y1))
        println("Rate of change: ",change)
        end

println("Enter the lower limit of the range")
x = parse{Int64,readline()}
println("Enter the upper limit of the range")
x=parse{Int64,readline()}
println("Enter the lower limit of the 2nd range")
y=parse{Int64,readline()}
println("Enter the upper limit of the 2nd range")
z=parse{Int64,readline()}

rateofchange(x,x,y,z)

Enter the lower limit of the range
stdin: 1
Enter the upper limit of the range
stdin: 2
Enter the lower limit of the 2nd range
stdin: 3
Enter the upper limit of the 2nd range
stdin: 4
Rate of change: 1.0
```



MY APPROACH AT TEACHING JULIA



TOOK MORE TIME
THAN EXPECTED



MADE SURE
STUDENTS HAD
ENOUGH
INFORMATION



ANSWERED
QUESTIONS



SEVERAL
OPPORTUNITIES TO
WRITE PROGRAMS

TEACHING JULIA

STUDENTS RESPONSE TO JULIA

Most enjoyed
learning something
new



```
graph TD; A[Most enjoyed learning something new] --> B[Did not like how precise coding was]; B --> C[wanted to learn more, but not take a programming class];
```

Did not like how
precise coding was

wanted to learn
more, but not take a
programming class

A red abstract graphic consisting of several overlapping, curved, ribbon-like shapes that sweep across the bottom left corner of the slide.

ACKNOWLEDGEMENTS

- NSF HBCU-UP Targeted Infusion Project Grant
- Julia Computing Diverse and Inclusion Grant
- Dr. Anna Harris
- Mr. Samuel Chaney

- " JuliaBox – Products – Julia Computing ." Julia Computing . 1 Jan 2012. Web. 11 Mar 2019. <<http://juliacomputing.com/products/juliabox.html>>

REFERENCES



QUESTIONS?

THANK YOU!

