

Programming Systems- Assignment 2

SILPA C

Ph.D.,IIIT Chittoor

February 9, 2017

Program to implement Towers of Hanoi Problem in JULIA

```
function toh(ndisks::Int64, startPeg, auxPeg, endPeg)
    if ndisks==1
        println("Move disk from peg $startPeg to peg $endPeg")
    else
        toh(ndisks-1, startPeg, endPeg, auxPeg)
        println("Move disk from peg $startPeg to peg $endPeg")
        toh(ndisks-1, auxPeg, startPeg, endPeg)
    end
end
toh(3, 'a', 'b', 'c')
```

OUTPUT:

```
julia> include("toh1.jl")
Move disk from peg a to peg c
Move disk from peg a to peg b
Move disk from peg c to peg b
Move disk from peg a to peg c
Move disk from peg b to peg a
Move disk from peg b to peg c
Move disk from peg a to peg c
```

Program to implement Insertion sort algorithm in JULIA

```
function isort(a,n)
    for i=1:n
        temp=a[i]
        j=i
        while j>1 && a[j-1]>temp
            a[j]=a[j-1]
            j-=1
        end
        a[j]=temp
    end
    println("$a")
end
```

OUTPUT:

```
julia> include("insert.jl")
isort (generic function with 1 method)

julia> isort([5,3,7,2,9,1],6)
[1, 2, 3, 5, 7, 9]
```

Program to implement queue operations in JULIA

```
type Que{T}
    a::Array{T,1}
end

Que() = Que{Any{}}
Que(a::DataType) = Que{a{}}
Que(a) = Que{typeof(a){}}

Base.isempty(q::Que) = isempty(q.a)

function Base.pop!{T}(q::Que{T})
    !isempty(q) || error("queue must be non-empty")
    pop!(q.a)
end

function Base.push!{T}(q::Que{T}, x::T)
    unshift!(q.a, x)
    return q
end

function Base.push!{T}(q::Que{Any}, x::T)
    unshift!(q.a, x)
    return q
end
```

OUTPUT: julia> include("que.jl")

```
julia> q=Que()
QueAny{Any{}}
```

```
julia> push!(q,1)
QueAny{Any[1]}
```

```
julia> push!(q,2.0)
QueAny{Any[2.0,1]}
```

```
julia> push!(q,"three")
QueAny{Any["three",2.0,1]}
```

```
julia> isempty(q)
```

```
false
```

```
julia> pop!(q)
```

```
1
```

```
julia> isempty(q)
```

```
false
```

```
julia> pop!(q)
```

```
2.0
```

```
julia> pop!(q)
```

```
"three"
```

```
julia> isempty(q)
```

```
true
```

```
julia> pop!(q)
```

```
ERROR: queue must be non-empty
```

```
in pop!(::QueueAny) at /home/silpa/Desktop/que.jl:15
```