

# PROCESSES VS THREADS DESIGN

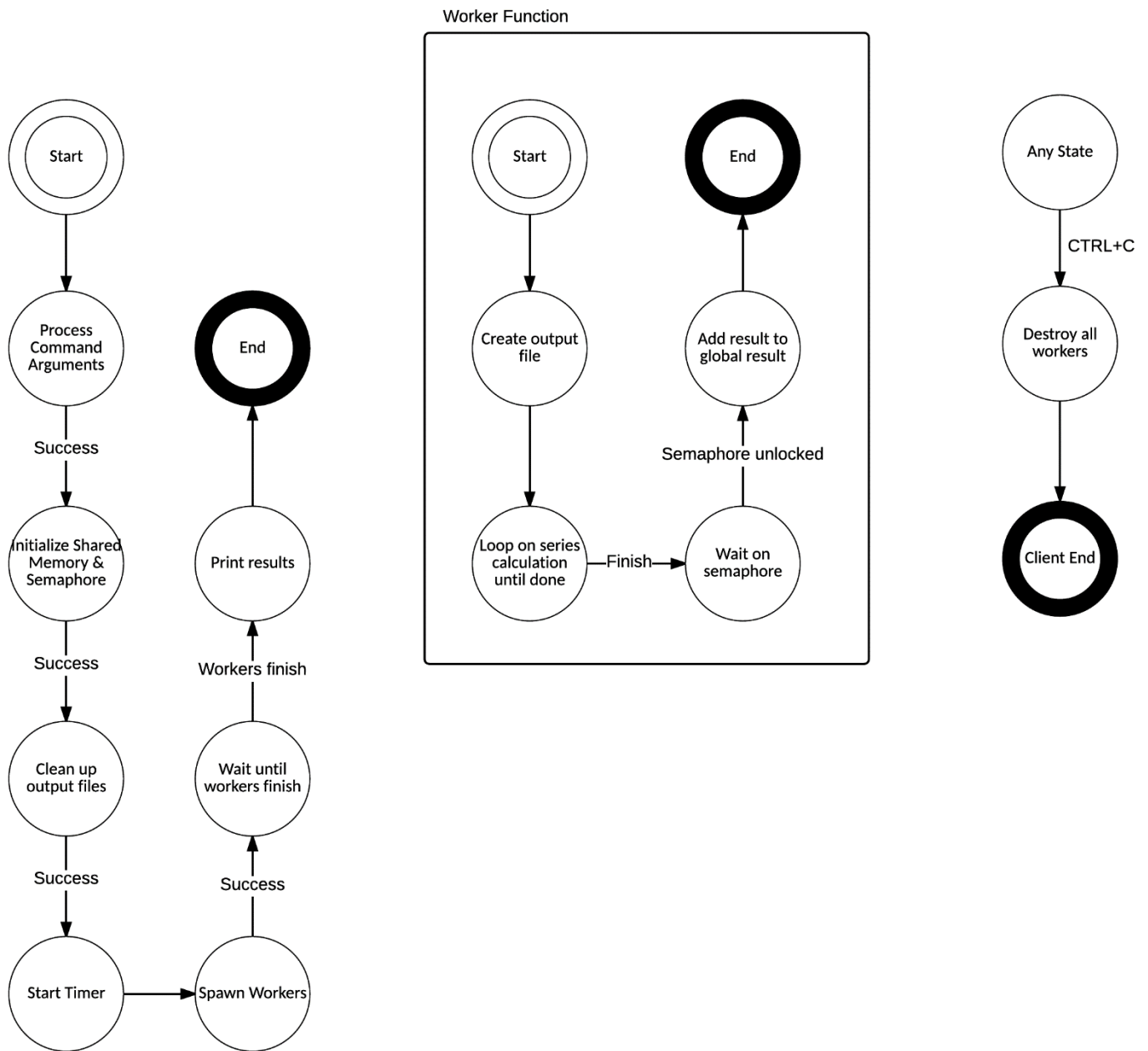
COMP 8005

SPENSER LEE A00925785

# Contents

State Chart Diagram .....	3
Pseudo Code .....	4

# State Chart Diagram



# Pseudo Code

## Process Command Arguments

- argp library parse call

## Initialize

- open name semaphore
- map global result variable
- initialize global result variable

## Clean Up Output

- if output is enabled
  - if output directory not created
    - create output directory
  - if results file exists
    - delete old output files

## Start Timer

- initialize timer
- set start time

## Spawn Workers

- totalworkers = number of process \* number of threads
- workeriterations = totaliterations / totalworkers

- start iteration = 0

- for each process

  - fork

  - if child

    - if threading

      - for number of threads per process

        - create thread of Worker Function

        - increment start iteration

      - for number of threads per process

        - join to child process

    - else

      - Worker Function

      - exit

  - if parent

    - if threading

      - increment start iteration by num iterations per worker \* num threads per worker

    - else

      - increment start iteration by num iterations per worker

**Worker Function**      **NOTE: this is where the Taylor Series is implemented**

```
if start is even
    sign = -1
else
    sign = 1
denominator = 3 + 2*start
create out file
for number of iterations
    result = result + (sign * (1/denominator))
    denominator += 2
    sign *= -1
wait for semaphore
lock semaphore
global result += result
unlock semaphore
```

**Wait Until Workers Finish**

```
wait for child processes to finish before continue
```

**Print Results**

```
set finish time
print 4 * (1 + global result)
print actual pi
print finish time - start time
```