Bachelor of IT (Computer Science)
Assignment 1
CAB301 - Algorithms and Complexity

Dane Madsen
n10983864@qut.edu.au

Contents

1 Algorithm Design and Analysis

1.1 FirstComeFirstServed Method

This method is used to sort jobs for first come first served scheduling. It achieves this by using the selection sort algorithm to sort the jobs by their arrival time.

```
ALGORITHM FirstComeFirstServed()
  // Returns a new array of jobs sorted by their arrival time
  $A \leftarrow Jobs.ToArray()$
  for $i \leftarrow 0$ in A.Length - 1 do
        for $j \leftarrow i + 1$ in A.Length do
              if $A[i].TimeRecieved > A[j].TimeRecieved$
                    $temp \leftarrow A[i]$
                    $A[i] \leftarrow A[j]$
                    $A[j] \leftarrow temp$
  return $A$
```

1.2 Priority Method

This method is used to sort jobs for priority scheduling. It achieves this by using the insertion sort algorithm to sort the jobs by their priority.

```
ALGORITHM Priority()
  // Returns a new array of jobs sorted by their priority
  $A \leftarrow Jobs.ToArray()$
  for $i \leftarrow 1$ in A.Length do $i++$
        for $j \leftarrow i$ in 0 do $j--$
              if $A[j].Priority < A[j-1].Priority$
                    $temp \leftarrow A[j]$
                    $A[j] \leftarrow A[j-1]$
                    $A[j-1] \leftarrow temp$
              else
                    break
  return $A$
```

1.3 ShortestJobFirst Method

This method is used to sort jobs for shortest job first scheduling. It achieves this by using the insertion sort algorithm to sort the jobs by their length.

```
**ALGORITHM** *ShortestJobFirst()*
  // Returns a new array of jobs sorted by their length
  $A \leftarrow Jobs.ToArray()$
  for $i \leftarrow 0$ in $A.Length - 1$ do
      for $j \leftarrow 0$ in $A.Length - i - 1$ do
          if $A[j].ExecutionTime > A[j+1].ExecutionTime$
              $temp \leftarrow A[j]$
              $A[j] \leftarrow A[j+1]$
              $A[j+1] \leftarrow temp$
  return $A$
```

2 Testing

2.1 Jobs ADT

2.2 JobCollection ADT

2.3 Scheduler ADT