Alejandro Chavez
Assignment 5 - Discrete Mathematics
November 14, 2013

Assignment 5

Chapter 3.1

- 4)

  ```
  procedure max difference($a_1, a_2, ..., a_n$ :integers)
  j:= 1
  k:= 2
  maxdiff := 0
  while (k < n+1)
      diff := $a_j - a_k$
      if diff > maxdiff
          maxdiff := diff
  ```

- 10)

  ```
  procedure exponentiate($x$ : floating point; $n$ : integer)
  if (n = 0)
      return 1
  k := 1
  if (n < 0)
      k := $-1$
  start := 1
  i := 1
  while (i<n+1)
      start := x$*$start
      i++
  return k$*$start
  ```

- 16)

  ```
  procedure smallest int($a_1, a_2, ..., a_n$ : integers)
  small = $a_1$
  j := 2
  while (j < n+1)
      if ($a_j$ < small)
          small := $a_j$
      j++
  return small
  ```

- 28)

  ```
  procedure create sublist($a_1, a_2, ..., a_n$ : integers)
  ```

```
objects := n
location := 0
for x := 1 to 4
  length := floor(objects/(4 − (x−1)))
  objects := objects−length
  for y := 1 to length
    L[x][y] := a[y+location]
  location := location + length
return L
```

- 50)

- 56)

Chapter 3.3

- 1)
  $n - 1$

- 2)

```
procedure inc. four terms($a_1, a_2, ..., a_n$ : integers)
for y:= 1 to 3
  steps:= 4−y
  i:= 1
  j:= 2
  for x:= 1 to steps
    while ($j <= n$)
      if ($a_i > a_j$)
        k:= $a_i$
        $a_i$ := $a_j$
        $a_j$ :=k
        j++
        i++
```

  This algorithm will always do the same number steps for any value of n except for the while statement for values of n less than 4, and thus has time complexity $O(1)$ in terms of the number of comparisons used..

- 4)

- 8)

- 16)