HW2:
Max: 47
Min: 21
Avg: 38.76

Distribution:

\[
\begin{array}{cccc}
21 & 39 & 39 & 37
\end{array}
\]
Midterm Oct 26 (Wed) 2005, in-class

Q6

\[ \frac{a}{b} \]

\[ c \text{ source} \]

\[ i \quad i+k \quad (k \geq 0) \]

- after adding node 6.

- incremental cost of adding c to b

\[ = j \]

- incremental cost of adding c to a

\[ = k \]

We have two cases - \( k > j \) or \( k < j \)

\[ \begin{align*}
\text{Bin} & \\
i+j < i+k & \quad i+k < j+j
\end{align*} \]

In both cases Bin & Min Eny cost free are same!
2. All answers & 2 also

3. In the worst case

\[
\frac{17}{2} \leq \frac{\text{Cost (BIP True)}}{\text{Cost (Opt True)}} \leq 12
\]

4. 

5. Instead of \( P_a + P_b \) energy to reach both \( a \) \& \( b \) for Node 1 just need \[ \max (P_a, P_b) \]
Q7

1) Traditional OS are very are
   1. Very bulky (in terms of code size, resources used)
   2. Not geared toward event-driven processing.

2) Why component-based?
   (a component offers functionality (or service) to other objects)
   Helps to optimize the code which needs to be downloaded or to store.
   E.g., a component which is not used by an application is not excluded from that application's executable.
3. distinguishing features of events & tasks in TinyOS.

- Events: Asynchronous

- Tasks are executed to completion W.R.T other tasks.
- Events can preempt the current task.

2 priority. Events have greater priority than tasks.

4. module & configuration?

- Implementation code
- Wiring (linking up) the components.
5. Component Blink (uses interface Timer)
   Start
   Event Fired ← event handler.

   Component Timer (provides interface Timer)
   Command Start +

   Interface Timer
   3
   Command Start +
   Command Stop
   Event Fired

   Hierarchy of Components
   Higher-level Component Blink
   Command
   Event
   Lower-level Component Timer
A component which uses interface X provides implementation for commands in X.

A component which uses an interface X provides event handlers for events in X.
MLBT

- **Lifetime of node** \( p \) in \( T \)
  
  \[
  L(T, p) = \frac{B(p)}{f(T, p)} \]

  \( B(p) \) \( \rightarrow \) battery power of node \( p \)

  \( f(T, p) \) \( \rightarrow \) maximum power level among all its children.

  \[
  f(T, p) = \max_{q \in \text{children}(p)} \text{wpq}
  \]

- **Lifetime of Tree** \( T \) is \( \min_{p \in T} L(T, p) \) over all the nodes in the tree.