CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST:

LAST:

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
01 Chop(a_1, ..., a_n; b_1, ..., b_n)  \\ input is 2 lists of n integers, n is a power of 2
02       if (n = 1)
03             return a_1 b_1
04       else
05             p = n/2
06             rv = Chop(a_1, ..., a_p, b_1, ..., b_p)
07             rv = rv + Chop(a_1, ..., a_p, b_{p+1}, ..., b_n)
08             rv = rv + Chop(a_{p+1}, ..., a_n, b_{p+1}, ..., b_n)
09             rv = rv + Chop(a_{p+1}, ..., a_n, b_1, ..., b_p)
10             return rv
```

1. (5 points) Suppose that $T(n)$ is the running time of Chop on an input array of length n. Give a recursive definition of $T(n)$. Assume that dividing the list in half takes $O(n)$ time.

2. (4 points) What is the height of the recursion tree for $T(n)$, assuming n is a power of 2?

3. (3 points) What is the amount of work (aka sum of the values in the nodes) at level k of this tree?

4. (3 points) How many leaves are in the recursion tree for $T(n)$? (Simplify your answer.)

CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST: | **LAST:**

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
01 Crunch(k,n)   \\ inputs are positive integers
02         if (n = 1) return k
03         else if (n = 2) return k^2
04         else
05                 half = ⌊n/2⌋
06                 answer = Crunch(k,half)
07                 answer = answer*answer
08                 if (n is odd)
09                         answer = answer*k
10                 return answer
```

1. (5 points) Suppose $T(n)$ is the running time of Crunch. Give a recursive definition of $T(n)$.

2. (4 points) What is the height of the recursion tree for $T(n)$? (Assume that n is a power of 2.)

3. (3 points) How many leaves are in the recursion tree for $T(n)$?

4. (3 points) What is the big-Theta running time of Crunch?

CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST:

LAST:

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
01 Process (a_1, ..., a_n: array of integers)
02     if (n = 1)
03         if (a_1 > 8) return true
04         else return false
05     else if (Process(a_1, ..., a_{n-1}) is true and Process(a_2, ..., a_n) is true)
05         return true
06     else return false
```

1. (3 points) If Process returns true, what must be true of the values in the input array?

2. (5 points) Give a recursive definition for $T(n)$, the running time of Process on an input of length n, assuming it takes constant time to set up the recursive calls in line 05.

3. (3 points) What is the height of the recursion tree for $T(n)$?

4. (4 points) What is the big-theta running time of Process?

CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST:

LAST:

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
01 Twiddle(a_0, ..., a_{n-1}))  \\ input is an array of n integers
02       if (n = 2 and a_0 > a_1)
03             swap(a_0, a_1)   \\ interchange the values at positions 0 and 1 in the array
04       else if (n > 2)
05             p = ⌊n/4⌋
06             q = ⌊n/2⌋
07             r = p + q
08             Twiddle(a_0, ..., a_q)   \\ constant time to make smaller array
09             Twiddle(a_{q+1}, ..., a_{n-1})   \\ constant time to make smaller array
10             Twiddle(a_p, ..., a_r)   \\ constant time to make smaller array
```

1. (5 points) Suppose that $T(n)$ is the running time of Twiddle on an input array of length n. Give a recursive definition of $T(n)$.

2. (4 points) What is the height of the recursion tree for $T(n)$, assuming n is a power of 2?

3. (3 points) What is the amount of work (aka sum of the values in the nodes) at level k of this tree?

4. (3 points) How many leaves are in the recursion tree for $T(n)$? (Simplify your answer.)

CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST: | **LAST:**

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
01 MyFunc(a_1,..., a_n)  \\ input is an array of n integers
02     for i := 1 to n − 1
03         min := i
04         for j := i to n
05             if a_j < a_min then min := j
06         swap(a_i, a_min)   \\ interchange the values at positions i and min in the array
```

1. (4 points) If the input is $10, 5, 2, 3, 8$, what are the array values after two iterations of the outer loop?

2. (4 points) Let $T(n)$ be the number of times that line 5 is executed. Express $T(n)$ using summation notation, directly following the structure of the code.

3. (4 points) Find an (exact) closed form for $T(n)$. Show your work.

4. (3 points) What is the big-theta running time of MyFunc?

CS 173, Fall 2015
Examlet 11, Part A

NETID:

FIRST:

LAST:

Discussion: Thursday 2 3 4 5 Friday 9 10 11 12 1 2

```
00 Kitty(a_1, ..., a_n) : list of n positive integers, n ≥ 2)
01      if (n = 2) return |a_1 − a_2|
02      else
03          bestval = 0
04          for k = 1 to n
05              newval = Kitty(a_1, a_2, ..., a_{k−1}, a_{k+1}, ... a_n)   \\ constant time to remove a_k
06              if (newval > bestval) bestval = newval
07          return bestval
```

1. (3 points) Describe (in English) what Kitty computes.

2. (5 points) Suppose that $T(n)$ is the running time of Kitty on an input list of length n. Give a recursive definition of $T(n)$.

3. (3 points) What is the height of the recursion tree for $T(n)$?

4. (4 points) How many leaf nodes are there in the recursion tree for $T(n)$?