Name: ____

The exam consists of seven problems; you only need to solve five. You *must* do problems 1, 2, and 3; the remaining two may be chosen from problems 4 - 7. Please indicate the problems you want to have graded by circling the problem numbers — otherwise, I will just grade the first four problems you worked on.

The following reference materials are provided on the last page of the exam: the statement of the Master Theorem and some summation formulas.

- You have 120 minutes.
- You may use only a calculator, pencil, or pen; you may not use a phone as a calculator.
- You must show all calculations. If you fail to show your work, you will receive no credit.
- You **must** put away all notes and papers and close all bags.

1. (**REQUIRED**) Consider the following algorithm to multiply two *n*-bit numbers x and y:

RECURSIVE-MULTIPLY(x, y)1 n = x.size2**if** n == 13 return xy4elseWrite $x = x_1 \cdot 2^{n/2} + x_0$ and $y = y_1 \cdot 2^{n/2} + y_0$ 5 $\mathbf{6}$ $p = \text{Recursive-Multiply}(x_1 + x_0, y_1 + y_0)$ 7 $q = \text{Recursive-Multiply}(x_1, y_1)$ $r = \text{Recursive-Multiply}(x_0, y_0)$ 8 9 return $q \cdot 2^{n} + (p - q - r) \cdot 2^{n/2} + r$

- (a) Derive a recursion for the running-time of RECURSIVE-MULTIPLY.
- (b) Solve the recursion to find an asymptotic bound for the running-time.
- (c) Show that RECURSIVE-MULTIPLY is equivalent to the 'schoolbook' method of multiplication.

2. (**REQUIRED**) Consider the following recursive function which computes the minimum value in an array x of length x. *length*:

```
RECURSIVE-MIN(x)

1 n = x. length

2 if n == 1

3 return x[1]

4 else

5 p = \text{RECURSIVE-MIN}(x[2..n])

6 return MIN(x[1], p) // two-argument MIN() function
```

(a) Derive a recursion for the running-time of RECURSIVE-MIN.

(b) Use a recursion tree to 'guess' an asymptotic bound for the recursion.

(c) Use the substitution method to prove your guess is correct.

3. (**REQUIRED**) Consider the following variant of the subset-sum problem. There are *n* items, each having non-negative, integer weight w_i , i = 1, 2, ..., n. We wish to select a set *S* of items such that

$$\sum_{i \in S} w_i \le W$$

where W is a non-negative, integer bound. Moreover, we want S to maximize the sum, subject to the constraint.

- (a) Show that this problem has optimal substructure.
- (b) Consider the following greedy approach: at each step, add to S the item of largest weight that maintains the bound $(\sum_{i \in S} w_i \leq W)$. Construct an example that shows that this greedy approach does not necessarily produce an optimal solution.

4. The algorithm P-SUM computes the sum of the elements of an array L of length n:

P-SUM(L) 1 n = L.length2 if n == 13 return L[1]4 $c = \lfloor n/2 \rfloor$ 5 x =spawn P-SUM($L[1 \dots c]$) 6 y =P-SUM($L[c + 1 \dots n]$) 7 sync 8 return x + y

- (a) Draw the computation DAG of P-SUM(L) for L of length 4.
- (b) Identify the critical path on the DAG; determine the span.
- (c) Determine the work and parallelism of P-SUM(L) with L of length 4.

- 5. Consider an RSA key set with p = 17, q = 19, n = 323, and e = 5.
- (a) Use the Extended Euclidean Algorithm to find the decryption exponent d.
- (b) Show that a = 2 is *not* a witness to the compositeness of q.
- (c) Verify that S = 4 is a valid signature for the message M = 55.

6. Consider the following flow graph G:



- (a) Show the execution of the Edmonds-Karp algorithm on G. For each iteration, show the flow f on the graph G, the residual graph G_f , and the augmenting path. Include all backwards flows in the residual graph.
- (b) What is the minimum cut corresponding to the maximum flow on this network?

7. Let G be an undirected graph, u and v vertices of G, and k a non-negative integer.

Define LONGEST-PATH-LENGTH(G, u, v) to be the optimization problem of determining the length of the longest simple path from u to v in G, and LONGEST-PATH(G, u, v, k) the decision problem "is there a simple path from u to v in G of length at least k?" Show that LONGEST-PATH $\in P$ if and only if LONGEST-PATH-LENGTH is solvable in polynomial time. Theorem (Summation Formulas).

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$
$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$
$$\sum_{i=0}^{n} r^i = \frac{1-r^{n+1}}{1-r}$$

Theorem (Master Theorem). Let $a \ge 1$ and b > 1 be constants, let f(n) be a function, and let T(n) be defined on the nonnegative integers by the recurrence

$$T(n) = aT(n/b) + f(n)$$

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lfloor n/b \rfloor$. Then T(n) has the following asymptotic bounds:

- (a) If $f(n) = O(n^{\log_b a \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
- (b) If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.
- (c) If $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, and if $af(n/b) \le cf(n)$ for some constant c < 1 and all sufficiently large n, then $T(n) = \Theta(f(n))$.