CS 2233 Discrete Mathematical Structures – Fall 09

11/9/09

7. Homework

Due **11/18/09** before class

Please refer to the corresponding exercise sections in the textbook (Rosen, 6th edition).

1. **Expansion Method** (2 points)
   For the recurrence below use the expansion method to find a guess of what it could solve to. Make your guess as tight as possible.
   $T(1) = 1$, and $T(n) = 2T(\frac{n}{2}) + n$ for $n \geq 2$.

2. **Master Theorem** (8 points)
   Use the master theorem to prove the following claims. Justify your answers.

   (a) (2 points) $T(n) = 3T(\frac{n}{3}) + 1$

   (b) (2 points) $T(n) = 27T(\frac{n}{3}) + n^5$

   (c) (2 points) $T(n) = 25T(\frac{n}{5}) + 5n^2$

   (d) (2 points) $T(n) = 8T(\frac{n}{2}) + n^2$

3. **Recursive Mystery** (4 points)
   Consider the following recursive code for $n \geq 1$:

```
int mystery(int n){
  if(n==0) return 1;
  if(n==1) return 1;

  for(int i=0; i<n; i++){
     for(int j=0; j<n; j++){
        print("hello");
     }
  }
  int a = mystery(n/9);
  int b = mystery(n/9);
  int c = mystery(n/9);
  return a+b;
}
```

   (a) (2 points) Set up a runtime recurrence for the runtime $T(n)$ of this recursive algorithm.

   (b) (2 points) Solve this runtime recurrence using the master theorem. *(Hint: Use the definition of the logarithm:* $x = \log_b a \quad \Leftrightarrow \quad b^x = a$.*)*

4. Section 7.2 (page 471)
   (6 points) 4c,d. Use the theorem we covered in class.