Recitation 12

Taylor Spangler

April 9, 2012

- (3 min max) Questions about last week's quiz?
- Questions about lecture / homework so far?
- MAPLE!!!!
- Induction:Let's try a strong induction proof.

Show that if $n \in \mathbb{N}$ then $12 \mid (n^4 - n^2)$.

1. **Base Case:**
   (a) $n = 1$: $1^4 - 1^2 = 0 = 12 * 0$ so $P(1)$ is true.
   (b) $n = 2 : 2^4 - 2^2 = 16 - 4 = 12 = 12 * 1$ so $P(2)$ is true.
   (c) $n = 3 : 3^4 - 3^2 = 81 - 9 = 72 = 12 * 6$ so $P(3)$ is true.
   (d) $n = 4 : 4^4 - 4^2 = 256 - 16 = 240 = 12 * 20$ so $P(4)$ is true.
   (e) $n = 5 : 5^4 - 5^2 = 625 - 25 = 600 = 12 * 50$ so $P(5)$ is true.
   (f) $n = 6 : 6^4 - 6^2 = 1296 - 36 = 1260 = 12 * 105$ so $P(6)$ is true
2. **Strong Inductive Hypothesis** Let $k \geq 6 \in \mathbb{N}$ and assume that $12 \mid (m^4 - m^2)$ for $1 \leq m \leq k$ where $m \in \mathbb{N}$.
3. Let $i = k - 5$, then we can assume $P(i)$ holds. Clearly $i + 6 = k + 1$.
4. $(i + 6)^4 - (i + 6)^2 = (i^4 + 24i^3 + 180i^2 + 864i + 1296) - (i^2 + 12i + 36) = (i^4 - i^2) + 24i^3 + 180i^2 + 852i + 1260$
5. Clearly $i^4 - i^2 = 12 * t$.
6. We can also see that the remaining portion can be rewritten as $12(2i^3 + 15i^2 + 71i + 105)$.
7. Then we have that $(k + 1)^4 - (k + 1)^2 = 12t + 12(2i^3 + 15i^2 + 71i + 105) = 12 * (t + 2i^3 + 15i^2 + 71i + 105)$
8. Then we have that $12 \mid (k + 1)^4 - (k + 1)^2$.

- Algorithms - Greedy algorithm for optimal change making:
  - Start with the largest value coin, then as the value remaining decreases below that of the largest value coin, begin using smaller values:
    - Example make change for 84 cents
    - Pick 25 we still have 59 cents
    - Pick 25, we still have 34 cents
    - Pick 25, we still have 9 cents
    - Can't pick 10, pick 5 we still have 4 cents
    - ...eventually picking 4 pennies.
    - Total of 8 coins
  - Is this algorithm correct (sound)? No, suppose we had a 21 cent value coin...this algorithm would give 8 coins (as above), but the optimal would be 4-21 cent coins. (it is sound if we do limit ourselves to the standard US coins though)
  - Is this algorithm complete? It will always give an answer, even if it has to use pennies
  - Is this algorithm finite? Yes, it will never take an infinite amount of steps.
- Algorithm 2e...let's write this algorithm using algorithm 2e

$C \leftarrow \{100, 25, 10, 5, 1\}$
$S \leftarrow \emptyset$ //cointains solution
$t \leftarrow 0$; //contains the sum of elts in S
while $t \neq n$ **do**
  | $x \leftarrow$ largest elt in C such that $x + t \leq n$ $S \leftarrow S \cup \{$coin of value $x\}$ $t \leftarrow t + x$
end
Return S

Algorithm 1: makeChange(n)

- Is this algorithm efficient in terms of the running time?
  - Running time:
  - So this requires at most n iterations through the loop. So one might think that this is linear.
  - However, the input to the function is n, which is of size $\log(n)$.
  - So since $n = 2^{\log(n)}$ then we get exponential time with respect to the input.