CS182 – Foundations of Computer Science

Homework 5 (100 points)

Due: Thursday, April 2, 2020, 11:59pm

Upload the homework to Gradescope. No late submissions accepted. Only typed solutions will be graded.

Problem 1. 20 points
The best way to prove this is by using Principle of Mathematical induction. In the following equations, % denotes the modulo operator ($a\%b$ gives the remainder we get when we divide a by b).
Firstly,

$$P(1) : (1^3 + 2.1 + 3)\%3 = (6\%3) == 0$$

is true. (== here is equivalent to checking if LHS equals RHS. It is an operation)

Assume P(k) is true, then, $(k^3 + 2k + 3)\%3 == 0$. Let us have, $(k^3 + 2k + 3) = 3\lambda$ where λ is some integer as it is a multiple of 3.
We have to prove,

$$P(k+1) : ((k+1)^3 + 2(k+1) + 3)\%3 == 0$$

(the equality should hold for $P(k+1)$.
Simplifying the LHS, we get,

$$P(k+1) : (k^3 + 1 + 3k^2 + 3k + 2(k+1) + 3)\%3$$

Rearranging the terms,

$$P(k+1) : (k^3 + 2k + 3 + 3k^2 + 3k + 3)\%3$$

which further simplifies to

$$P(k+1) : (3\lambda + 3(k^2 + k + 1))\%3 = 0$$

Hence $P(k+1)$ is proved. Thus, by Mathematical induction, we have proved that $(n^3 + 2n + 3)$ is divisible by 3.

Problem 2. 20 points
We will use Mathematical induction for this problem as well.
Firstly, $P(1) : \sum_{i=1}^{i=1} i.2^i = 1.2^1 = 2 \leq 4 = 2^{1+1}.1$ is true.

Assume P(k) is true, then,

$$P(k) : \sum_{i=1}^{i=k} i.2^i \leq (k)2^{k+1}$$

.

We have to prove,

$$P(k+1) : \sum_{i=1}^{i=k+1} i.2^i \leq (k+1).2^{k+2}$$

(the inequality should hold for $P(k+1)$.
Simplifying the LHS, we get,

$$P(k+1) : \sum_{i=1}^{i=k} i.2^i + (k+1)2^{k+1}$$

Now, using our assumption, we get,

$$P(k+1) : \sum_{i=1}^{i=k} i.2^i + (k+1)2^{k+1} \leq (k)2^{k+1} + (k+1)2^{k+1}$$

$$P(k+1) : k.2^{k+1} + 2^{k+1} + (k)2^{k+1} \leq k.2^{k+2} + 2^{k+1} \leq k.2^{k+2} + 2^{k+2} = 2^{k+2}(k+1)$$

. Thus, by Mathematical induction, we have proved the problem statement.

Problem 3. 20 points

Firstly, $P(1) : \sum_{i=1}^{i=2} 1/i = 1 + 1/2 = 3/2 \geq 1/4$ is true.
Assume P(k) is true, then,

$$P(k) : \sum_{i=1}^{i=2^k} 1/i \geq k/4$$

.

We have to prove,

$$P(k+1) : \sum_{i=1}^{i=2^{k+1}} 1/i \geq (k+1)/4$$

(the inequality should hold for $P(k+1)$.
Simplifying the LHS, we get,

$$P(k+1) : \sum_{i=1}^{i=2^k} 1/i + 1/(2^n + 1) + +1/(2^n + 2) + ... + 1/(2^{n+1}) \geq k/4 + 2^n.(1/2^{n+1})$$

$$P(k+1) : k/4 + 2 = (k+2)/4 \geq (k+1)/4$$

which is true, so $P(k+1)$ is true. Thus, by Mathematical induction, we have proved the problem statement.

Problem 4. 20 points
We will use strong induction here.
Firstly, $P(1) : a_1 = 2 \leq 3^1$ is true. Also, $P(2) : a_2 = 9 \leq 3^2$ and thus $P(2)$ is true.
Assume the hypothesis is true for all numbers i smaller or equal to k. Then,

$$P(i) : a_i \leq 3^i$$

is true if i is smaller or equal to k.
We have to prove,

$$P(k+1) : a_{k+1} = 2a_k + 3a_{k-1} \leq 3^{k+1}$$

(the inequality should hold for $P(k+1)$.
Simplifying the LHS and using our assumptions, we get,

$$P(k+1) : a_{k+1} \leq 2.3^k + 3.3^{k-1} = 3.3^k = 3^{k+1} \leq 3^{k+1}$$

which is true, so $P(k+1)$ is true. Thus, by strong induction, we have proved the problem statement.

Problem 5. 20 points
The most efficient way is to use change of base rule to make it as $m^{constant}$. Another way is to use exponentiation by squaring. But it is not as fast as using change of base. We describe only the method using change of base here. We claim that $log_5 m = m^c$ where c is a constant. Proof is as follows. Firstly, $log_5 m = log_2 m / log_2 5 = c.log_2 m$ where c is a constant. Thus,

$$5^{log_2 m} = 5^{c log_5 m} = (5^{log_5 m})^c = m^c$$

. m^c can be computed in constant time. Thus, this the most efficient algorithm (you cannot get number of steps less than a constant considering the time required for accepting input). The algorithm is presented below.

```
procedure findexp (m){
    c = 1/log (5,2);
    return m^c;
}
```

In the above code, assume that log(a,b) returns $log_b a$.