Name: ____________________

Entry number: ____________________

There are 2 questions for a total of 10 points.

---

1. Solve the following problems. Show the working in the space provided.

   (a) (1 point) What is the last digit of 7^{100}?

   (a) ____1____

   Solution: $7^{100} \equiv (7^2)^{50} \equiv (-1)^{50} \equiv 1 \ (mod\ 10)$. So, the last digit of 7^{100} is 1.

   (b) (1 point) What is the value of $(2^{100} \cdot 3^{60}) \ (mod\ 5)$?

   (b) ____1____

   Solution: $2^{100} \equiv (2^2)^{50} \equiv (-1)^{50} \equiv 1 \ (mod\ 5)$ and $3^{60} \equiv (3^4)^{15} \equiv (1)^{15} \equiv 1 \ (mod\ 5)$. So, $2^{100} \cdot 3^{60} \ (mod\ 5)$ is equal to 1.

   (c) (1 point) What is the remainder when $\sum_{i=1}^{100}(i)!$ is divided by 9?

   (c) ____0____

   Solution: For every $k \geq 6$, we have $k!$ is divisible by 9. So, $\sum_{i=1}^{100} i! \equiv (1! + 2! + 3! + 4! + 5!) \ (mod\, 9)$. Since $1! \ (mod\, 9) = 1, 2! \ (mod\, 9) = 2, 3! \ (mod\, 9) = 6, 4! \ (mod\, 9) = 6, 5! \ (mod\, 9) = 3$, we have $\sum_{i=1}^{100} i! \ (mod\ 9) = (1 + 2 + 6 + 6 + 3) \ (mod\ 9) = 0$.

   (d) (2 points) Prove or disprove: $(2^n + 6 \cdot 9^n)$ is divisible by 7 for every $n \geq 0$.

   Solution: We will prove the statement using modular arithmetic. Note that $2^n \equiv 9^n \ (mod\ 7)$. So, we have $(2^n + 6 \cdot 9^n) \equiv (9^n + 6 \cdot 9^n) \equiv (7 \cdot 9^n) \equiv 0 \ (mod\ 7)$. Hence, $(2^n + 6 \cdot 9^n)$ is divisible by 7 for all $n \geq 0$.

2. (5 points) Consider the following problem:

   HALTING-INPUT: Given the description $\langle A \rangle$ of algorithm A, determine if there is a halting input for A (that is, there exists an input on which A halts).

   An algorithm P is said to solve the above problem if $P(\langle A \rangle)$ halts and outputs 1 when A has a halting input, and it halts and outputs 0 otherwise.

   Prove: There does not exist an algorithm P that solves the problem HALTING-INPUT.

Solution: In the class we have shown that there does not exist an algorithm for the halting problem.

HALTING: Given the description $\langle A \rangle$ of an algorithm A and an input x, determine if A halts on input x.

We will prove the statement by contradiction. We will argue that if there exists an algorithm for the above problem, then there also exists an algorithm for the halting problem. This implies that such an algorithm cannot exist. Indeed, suppose for the sake of contradiction there exists an algorithm P that determines if a given algorithm has a halting input. We will construct an algorithm Q for the halting problem that uses P as a subroutine. First, we need to define an algorithm B_A with respect to another algorithm A and input string x.

$B_{A,x}(y)$
- if $(x \neq y)$ then loop infinitely
- else execute $A(y)$

Here is the description of the algorithm Q for the halting problem.

$Q(\langle A \rangle, x)$
- return$(P(\langle B_{A,x} \rangle))$

It would be sufficient to show that Q outputs 1 iff A halts on x.

Claim 1: $Q(\langle A \rangle, x) = 1$ iff A halts on x.

Proof. From definition of $B_{A,x}$, it does not halt on any input $y \neq x$. So, $B_{A,x}$ has a halting input iff A halts on x. This implies that $Q(\langle A \rangle, x)$ outputs 1 iff A halts on x. □