University of Puerto Rico Mayagüez
Department of Electrical and Computer Engineering
ICOM 4075: Foundations of Computing

Partial Exam 2 (100 Points)

Name: CLAVE　　　　**Section:** ____________

Instructor: B. Vellez

You will have 120 minutes to answer as many questions as possible. Any points that you accumulate beyond 100 will be considered bonus points that you can accumulate towards improving your grades in other partial exams.

Question 1 (15 points). Show that if a and b are positive integers, then

$$ab = gcd\ (a, b) \cdot lcm(a, b)$$

HINT: Use the Fundamental Theorem of Arithmetic and the definitions of gcd(a,b) and lcm(a,b) in terms of the prime factors of a and b.

By the Fundamental Theorem of Arithmetic both a and b can be written as a product of primes.

$$a = \prod_{i=1}^{n} p_i^{M_i} \qquad b = \prod_{i=1}^{n} p_i^{l_i}$$

p_i's are prime factors. M_i's & l_i's their powers. 7/0.

Also we know:

$$gcd\ (a,b) = \prod_{i=1}^{n} p_i^{\min(M_i, l_i)} \qquad lcm\ (a,b) = \prod_{i=1}^{n} p_i^{\max(M_i, l_i)}$$

But $\min(M_i, l_i) + \max(M_i, l_i) = M_i + l_i$

$$gcd(a,b) \cdot lcm(a,b) = \prod_{i=1}^{n} p_i^{\min(M_i,l_i)} \cdot \prod_{i=1}^{n} p_i^{\max(M_i,l_i)} = \prod_{i=1}^{n} p_i^{\min() + \max()}$$

$$= \prod_{i=1}^{n} p_i^{M_i + l_i} = \prod_{i=1}^{n} p_i^{M_i} \cdot \prod_{i=1}^{n} p_i^{l_i} = a \cdot b.$$

Q.E.D.

Question 2a. (10 points). Show that Z_m (m>1) with addition modulo m ($+_m$) satisfies the associative property,

$$(a +_m b) +_m c = a +_m (b +_m c)$$

Useful definitions and theorems to remember:

$a +_m b = (a + b) \bmod m$

For all a in Z_m, a = a $\bmod m$

$(a + b) \bmod m = (a \bmod m + b \bmod m) \bmod m$

$(a +_m b) +_m c = ((a+b) \bmod m) +_m c$

$= (((a \bmod m + b \bmod m) \bmod m + c) \bmod m$

$= ((a \bmod m + b \bmod m) \bmod m + c \bmod m) \bmod m$

$= (a \bmod m + b \bmod m + c) \bmod m$

$= (a \bmod m + (b \bmod m + c \bmod m)) \bmod m$

$= (a \bmod m + (b \bmod m + c \bmod m) \bmod m) \bmod m$

$= (a \bmod m + (b+c) \bmod m) \bmod m$

$= (a \bmod m + (b +_m c)_{Z}) \bmod m$

$= (a + (b +_m c)) \bmod m = a +_m (b+c)$

Question 2b. (5 points). Assert what member of Z_m is the identity element and for any element **a** in Z_m and what member of Z_m is the inverse of **a** with the operation $+_m$.

Identity element of $+_m$ = 0

Additive inverse of element a in Z_m = $-a \bmod m = (m-a) (\bmod m)$

Question 3 (10 points). Show that if a, b, c and m are integers such that m>1 and c>0 and then

$$a \equiv b(mod\ m) \rightarrow ac \equiv bc(mod\ mc)$$

Direct Proof:

Let a, b, c arbitrary integers and m an arb. positive integer

Assume $a \equiv b \pmod{m}$

$\rightarrow a-b = mk$ for some integer k.

Then $c(a-b) = (cm)k = ac - bc$

Therefore $ac \equiv bc \pmod{mc}$

Q.E.D.

Question 4a. (5 points). Find a recurrent relation for the n[th] element of the sequence $\{a_n\}$ starting with a_1=2 and continuing as follows:

$$2, 5, 11, 23, 47, \ldots$$

$$a_n = \underline{(a_{n-1}) \cdot 2 + 1}$$

$$a_1 = 2$$

Question 4b. (5 points). Find the following element of the sequence satisfying the recurrence relation from part a:

$$a_8 = \underline{(a_7)2+1}$$

$$= ((a_6)2+1)2+1$$

$$= 4a_6 + 2 + 1$$

$$= 2^2 a_6 + \sum_{i=1}^{2} i$$

$$\vdots$$

$$= 2^7 a_1 + \sum_{i=1}^{7} i$$

$$= 2^8 + \underbrace{\frac{7(8)}{2}}_{\text{Euclid}} = 256 + 28 = \boxed{284}$$

$a_5 = 47$

$a_6 = 95$

$a_7 = 191$

$\boxed{a_8 = 383}$

$$\begin{array}{r} 191 \\ 2 \\ \hline 382 \end{array}$$

Question 5 (15 points). Find a __closed__ solution or formula for the following recurrence relation:

$$a_0 = 4$$
$$a_n = a_{n-1} + 2n + 3$$

You may use either forward or backward substitution. Remember that a closed solution may not be expressed recurrently.

$$a_n = \underline{(n+2)^2}$$

$a_0 = 4$

$a_1 = a_0 + 2\cdot 1 + 3$

$= 4 + 2 + 3$

$a_2 = (4+2+3) + 2\cdot 2 + 3$

$= 2(1+2) + 2\cdot 3 + 4$

$a_3 = (2(1+2) + 2\cdot 3 + 4) + 2\cdot 3 + 3$

$= 2(1+2+3) + 3\cdot 3 + 4$

$\vdots$

$a_n = 2\sum_{i=1}^{n} i + 3\cdot n + 4$

$= \cancel{2}\left(\frac{n(n+1)}{\cancel{2}}\right) + 3n + 4$

$= n^2 + n + 3n + 4$

$= n^2 + 4n + 4$

$= (n+2)^2$

Question 6a (10 points). Design a O(n) algorithm to determine if a list of integers is an arithmetic progression. The procedure should return the common difference of the progression or zero if the sequence is not a progression. In any case the returned value should be an integer number.

Remember that an arithmetic progression is a sequence defined by a recurrence of the form

$$a_n = a_{n-1} + d$$

where a_0 can be any initial number and d is a constant called the common difference. For instance, the sequence 4, 7, 10, 13 ... is an arithmetic progression with common difference 3.

```
procedure commonDifference(a1, a2, …, an: positive integers)
    if n<3 return Ø.
    i=3
    commonDiff = a2 - a1
    while i < n
        if ((ai - ai-1) ≠ commonDiff) return Ø.
        i=i+1
    return commonDiff.
```

Question 6b (5 points). Explain why your algorithm is O(n) Worst case analysis

In the worst case the sequence is an arithmetic progression and we must check every pair of consecutive elements. In this case the while repeats n-2 times

$(n-2) \in \Theta(n)$

Question 7a (10 points). Design a $O(n^2)$ procedure named PRP to determine if a list of positive integers is pairwise relatively prime. A list of integers is pairwise relatively prime if for each pair of integers a,b in the list, gcd(a,b)=1. You may assume that gcd(a,b) is an available function. The PRP procedure may only return either true or false.

```
procedure PRP(a₁, a₂, …, aₙ: list of positive integers)
```

for $i := 1$ to n

for $j = i+1$ to n

if $(\gcd(a_i, a_j) \neq 1)$ return false

return true.

Question 7b (5 points). Explain why your algorithm is $O(n^2)$ Worst case analysis.

In the worst case the sequence is PRP.

In this case both loops will execute exhaustively.

$$\text{Total calls to gcd} = \sum_{i=1}^{n} \sum_{j=i+1}^{n} = (n-1)+(n-2)+\ldots+1$$

$$= \frac{(n-1)n}{2} \in O(n^2)$$

Consider the following algorithm:

```
procedure mistery(n:positive integer > 0)
      result := 0
      i := 1
      while i<n
            result = result + 1
            i := i * 2
      return result
```

Question 8a (5 points) Express the value returned by `mystery` as a mathematical function of n (Hint: You may need to use floor and or ceiling)

We first evaluate mistery on a few points

n	mistery (n)
1	0
2	1
3	2
4	2
5	3

$$\text{mistery}(n) = \lceil \log_2 n \rceil$$

Question 8b (5 points) Calculate the number of arithmetic operations (*'s and +'s) as a function of n

The while loop runs $\lceil \log_2 n \rceil$ and each iteration does one addition + 1 multiplication.

$$\text{Total ops} = 2\lceil \log_2 n \rceil$$

Question 8c (5 points) Determine the asymptotic runtime complexity for the number of arithmetic operations of `mystery` using Big O notation for a given input n.

$$\Theta(\log n)$$