Staple!

Team C____

NT-Cryptography
MAT4930 *7554*

Home-C

Prof. JLF King
Touch: 2Jul2018

Due **BoC, Monday, 07Apr2014**. Please *fill-in* every *blank* on this sheet.

C1: *Show no work. Write* **DNE** *in a blank if the described object does not exist or if the indicated operation cannot be performed.*

a) Posints $K=$ ____, $N=$ ____, $\alpha=$ ____, $\beta=$ ____, are st. $\alpha \equiv_K \beta$, yet $N^\alpha=$ ____ is **not** $\equiv_K$ to $N^\beta=$ ____.

b) Using dictionary 0: ε, 1: "1", 2: "0", compute EnZiv(11110110)= ________________, in ⟨**7**⟩1⟨**34**⟩0... notation. In bits, EnZiv(11110110) is ________________.

OYOP: *Your 2 essay(s) must be* TYPED, *and Double or Triple spaced. Use the* Print/Revise *cycle to produce good, well thought out, essays. Start each essay on a new sheet.*

Do **not** *restate the problem; just solve it.*

C2: Let $\vec{\mathbf{1}} := 1111\ldots$, the half-$\infty$ constant-1 bit-string. Using our Ziv-algorithm, with dictionary that [initially] only has the nullword, we start parsing $\vec{\mathbf{1}}$.

Let $P(k)$ be the largest-number of bits we've parsed, having used-up at most k many bits from $\vec{\mathbf{1}}$. I.e, we Ziv-parse, and we eventually parse a new word [which we enter into our dictionary], having read exactly $P(k)$ many bits, in total, where $P(k) \leqslant k$. As we scan for the next new word, we run past the k^{th}-bit in $\vec{\mathbf{1}}$.

i) Give an approximate formula for $N(k)$, the number of words you've parsed, having read the first $P(k)$ many bits.

ii) Let $Z(k)$ be the length of the Ziv-compressed bit-string that encodes the first $P(k)$ many bits in the source-string. When k is large, give a pretty good estimate for $Z(k)$; a "closed formula", neither having a $\sum$ summation operator, nor a $\prod$ product operator.

What are approximate values for $N(500,000)$, and for $Z(500,000)$?

Compute $\lim_{k\to\infty} \frac{Z(k)}{k}$.

C3: Consider posreals $p+q=1$. Your coin outputs bit 0 with prob.$=p$, and bit 1 with prob.$=q$. Flipping the coin K times, the WLLN [Weak Law of Large Numbers] says, when K is "large", that a typical sequence has about pK many 0s, and has about qK many 1s.

α) Let $f(K)$ denote the number of such length-K bit-sequences. Estimate $f(K)$ using a binomial coefficient.

Now use Stirling's formula to get an "algebraic" estimate for $f(K)$ that just uses multiplication, division, and powers; it does *not* use factorials.

β) Define a fnc g by: $\boxed{2^{[K\cdot g(K)]} = f(K)}$. Using your "algebraic" formula for f, derive a formula estimating $g(K)$.

Assuming that all of your estimates could be proved rigorously, compute $\lim_{K\to\infty} g(K)$. What familiar formula is this limit?

γ) Using these ideas, do something extra. Impress me. [Hopefully, not with a brick...]

End of Home-C

C1:	___ ___	40pts
C2:	___ ___	85pts
C3:	___ ___	95pts
Total:	___ ___ ___	220pts

Honor Code: *"I have neither requested nor received help on this exam other than from my team-mates and my professor (or his colleague)."* *Name/Signature/Ord*

________________ Ord: ____

________________ Ord: ____

________________ Ord: ____