COSC 2001(A) 3.0—Fall 2000

COSC 2001A 3.0—Fall 2000

Posted: Dec. 15, 2000

Due: January 10th, 2001, *any time on that date*, in the COSC2001 A box.

ALTERNATE Problem Set No. 3—*For Section A only.*

This Problem Set # 3 is for those, and *only those*, students who were unable to fulfil their course requirements in COSC 2000A as a result of the Labour dispute.

It provides an ***alternative*** opportunity to complete required course work, and provides alternative extended deadlines ("regular" #3 was due Dec. 14th, 2000).

Papers ***must*** **be typed or word-processed (the "*must*" does not apply to diagrams)**, and deposited in a course drop-box on the due date.

► **Due time:** SEE ABOVE. **The Box will be cleared the following morning. Location of the drop-box:** There is box labelled 2001A on the first floor of CCB, in the corridor that leads to the Ariel Lab.◄

In this ALTERNATIVE Problem Set it is still allowed—but not required!—to submit **ONE joint paper that has a total of TWO co-authors from the same section**. The same mark, as assigned to such a joint paper, will be given to **each** of its two authors.

► **IFF** you are submitting ALTERNATE Problem Set #3 ***with*** a partner, then you ***must*** notify me (in the usual manner) as described below, **Prtnr1.**–**Prtnr4.:**

Prtnr1. Make a file called "partner" (no quotes). [Please do ***not*** call it "Partner" or "PARTNER" or "a3partner" or anything other than "partner"].

Prtnr2. Put in it your name and "ariel" login, ***and*** the name and ariel login of your partner as well.

Prtnr3. Give the following command on ariel

"**submit 2001 a3alt partner**"

Prtnr4. Only ***one*** submission (**Prtnr3.**, above) ***per pair*** please! ◄

If you do NOT plan to work with a partner please do NOT submit any co-author information!

General Remark. Each solution must contain *adequate explanation(s)* of *why* it answers the relevant question. While examples can help one to understand your point of view, *they are NOT substitutes* for a logical argument that establishes your solution's validity ***in general***.

1. Prove that a CFG with productions restricted to be of the two types below

$$A \to Ba$$

$$A \to a$$

produces a REGULAR language.

It is ***NOT*** allowed to use the result that regular languages are closed under reversal.

2. (a) Define a CFG G such that $L(G) = \{0^i 1^j : i < j\}$.

(b) **Convert** the grammar into a PDA, shown a **state diagram**, that **recognizes** $L(G)$ **by empty stack**.

No credit will be given to a "direct, brute force" PDA solution.

3. Code as simply as you can, *in state diagram form*, a TM which computes the function

$$f(x) = \begin{cases} 4 & \textbf{if } x = 0 \\ \uparrow & \textbf{otherwise} \end{cases}$$

I/O conventions *must* be the ones we adopted in class for Computability. Refer to the Web notes, Part I! You will need to specify, among other things, what is your tape alphabet.

(Recall that "$\uparrow$" stands for "undefined".)

4. Prove that $\{x : \phi_x = \lambda x.13\}$ is not recursive. Then draw the conclusion from this result (but NOT through a different argument!) that there are infinitely many different TMs that compute the constant function $\lambda x.13$.

5. Prove that the problem "Does the function computed by M_x have exactly two members in its *range*?" is unsolvable.