**HW**

1. Identify one function that can be represented completely in tabular form. Show the table.

The boolean operations AND, OR, and XOR can be represented completely in tabular form as shown below:

Oper1 Oper2 Oper1&&Oper2 Oper1||Oper2 Oper1^Oper2

true true true true false

true false false true true

false true false true true

false false false false false

1. Identify one function whose outputs can be described as an algebraic expression involving their inputs. Show the algebraic expression.

The computation of the area of a circle can be described as an algebraic expression involving the input radius:

area=3.14\*radius\*radus

1. Identify a function that cannot be described in terms of an algebraic formula.

The function which finds the minimum of 3 integers cannot be described in terms of an algebraic formula.

**HW**

1. Apply the TM M described in this handout, starting with the following initial status:

\* 1 1 0 \*

State=START r/w head

Show all the states of the TM leading up to the halt state.

\* 1 1 0 \*

State=ADD

\* 1 1 1 \*

State=RETURN

\* 1 1 1 \*

State=HALT

1. Describe a TM that replaces a string of 0s and 1s with a single 0. You can assume the TM starts with a legal number on the tape. Note that there is no need for the machine to erase everything. You just need to make sure the machine halts with \*0\* immediately to the left of the r/w head. Give your answer using the graphical notation.

(1/\*/R)

2

(1/0/L) (0/\*/R) (0/0/R)

START

1

HALT

3

(\*/\*/L) (0/0/L) (\*/\*/R)

1. What function does the following TM compute?

Current state current cell content value to write direction to move new state to enter

START \* \* left SUBTRACT

SUBTRACT 0 1 left BORROW

SUBTRACT 1 0 right RETURN

BORROW 0 1 left BORROW

BORROW 1 0 right RETURN

BORROW \* \* right ZERO

ZERO 0 0 right ZERO

ZERO 1 0 right ZERO

ZERO \* \* no move HALT

RETURN 0 0 right RETURN

RETURN 1 1 right RETURN

RETURN \* \* no move HALT

You must also show the graphical notation of this TM.

(0/1/L) (0/0/R)

(1/0/R)

HALT

BORROW

zero

(0/1/L) (\*/\*/R) (\*/\*/N)

(1/0/R)

SUBTRACT

START

(\*/\*/L)

RETURN

(1/0/R)

(\*/\*/N)

(0/0/R)

(1/1/R)

This TM decrements the value on the tape if it is >0, or leaves the value unaltered if it is 0

**HW**

1. Show that the statement invert X (whose action is to convert the value of X to zero if its initial value is nonzero and to 1 if its initial value is zero) can be simulated by a Bare Bones program segment. That is, write a program sequence in Bare Bones that simulates the action of the statement:

if X not 0:

X=0

else:

X=1

AUX=1

while X not 0:

X=0

AUX=0

while AUX not 0:

X=1

1. Show that the if-else structure can be simulated using Bare Bones. That is, write a program sequence in Bare Bones that simulates the action of the statement:

if X not 0:

S1

else:

S2

where S1 and S2 represent arbitrary statement sequences. Hint: use the invert operation.

AUX=X

while AUX not 0:

S1

clear AUX

AUX=X

INVERT AUX

while AUX not 0:

S2

clear AUX

1. Describe the function computed by the following Bare Bones program assuming the function’s input is represented by X and its output by Z

clear Z

while X not 0:

incr Z

incr Z

decr X

Z=2\*X and X=0

1. Write a program in Bare Bones that assigns 0 to Z if X is even; otherwise assign 1 to Z.

clear Z

AUX=X

while AUX not 0:

invert Z

decr AUX

1. Show that even our simple BareBones language contains more statements than necessary by showing that the clear statement can be replaced with combinations of other statements in the language:

Clear X can be done by:

while X not 0:

decr X