CSC520 Q242 Final Exam

***Part A***

Due by the end of this class

Total: /50

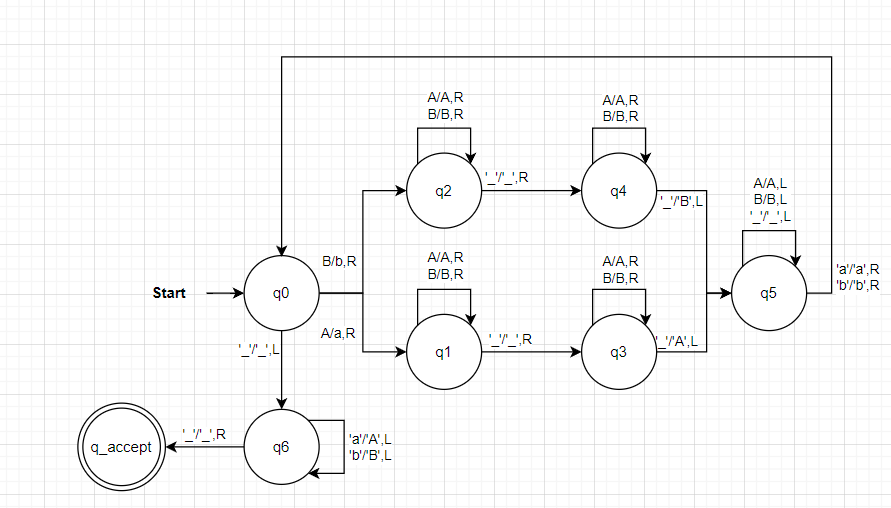
Name (last, first): Hassan, Sunzid

*“If you wish to understand you must …”*

# [20 points] Draw a state diagram for defining a Turning Machine to copy a given string. The given string can only contain ‘A’ or ‘B’, i.e. Z = {‘A’, ‘B’}. There need to be one (and only one) blank \_ between the given string and its copy. The result of the copied string can be located to the right or left of the given string.

# e.g. given in tape: AABBBA (the read write head is at the leftmost A, i.e. rwh = 0) (The input string can be of any length. If \_ is given, \_ will result.) Need to have AABBBA\_ AABBBA in the tape after the TM halts. (There may be any blank \_ before or after)

# (Your part B question 1 needs to be based on this state diagram.)



# [15 points] solvable and definable: (a) How many problems are definable on natural number N? (b) How many of those definable problems are solvable?

**Answer (a):**

Since all computable function *f*: N ⇒ N can be enumerated, the cardinality of the class of **all computable** functions is at most

*N*0

However, the cardinality of the class *P* of **all definable function** on N, where *F* = {*f*:(*f*: N⇒N)} is

2*N*0

**Answer (b):**

**Rice's Theorem (solvable case)**

Let *F* be any set of computable functions.

The set *S* = {*x*: *fx* F} is decidable ⇔ *F* = ⦰ or *F* is the set of all computable functions.

**Only trivial property of computable functions are decidable.**

EG: all function for one binary variable

EG2: all functions for two binary variables

# [15 points] *Complexity*: (a) Describe how to determine a problem is “less complex” than another problem. (b) Define “complexity degree”.

**Answer (a):**

**Complexity measure**

Measure relative complexity of computable function, without referring to concrete computational model.

**Complexity measure** assigns an appropriate resource bound *fe* to each computable function Ae

It does not specify the machine model or the resource but define the constraints with (Blum) axioms:

(A1) for each *e*, *fe* is defined on precisely the same domain as *Ae*

(A2) it is decidable form *e*, *n* and *r*, whether *fe* (*n*) == *r*.

**Complexity classes** of computable functions can be defined in terms of a total computable function g.

*C*(g) := {Ae : *n*f*e*(*n*) *g*(*n*)}

*C*(*g*) is the set of all computable functions with a complexity less than or equal to g.

*C*0 (*g*) := {*h* ∈ *C*(*g*): codom(*h*) {0,1}}

*C*0(*g*) is the set of all Boolean-valued functions with complexity less than *g*.

Using them as characteristic function on sets, *C0(g)* can be considered of as a complexity class of sets.

Ordering functions

*f* *g*

f grows no faster than g

where f and g be function from nonnegative integers to positive real numbers, and for some real constant c > 0, and some nonnegative integer constant *n*0, define

*f* ∈ *O*(*g*), (*f* *g*) f is upper bounded by g

*f* ∈ Ω(*g*), (*f* *g*) f is lower bounded by g

*f* ∈ Θ(*g*), (*f* = *g*) f is order g

*O*(*g*) := {*f* : ∀*n*>=*n0*(*f*(*n*) (*c* \* *g*(*n*)))}

*o*(*g*) := {*f* : ∀*n*>=*n0*(*f*(*n*) < (*c* \* *g*(*n*)))}

Ω (*g*) := {*f* : ∀*n*>=*n0*(*f*(*n*) (*c* \* *g*(*n*)))}

(*g*) := {*f* : ∀*n*>=*n0*(*f*(*n*) > (*c* \* *g*(*n*)))}

*Θ(g) := O(g) Ω(g)*

Each set (theta)(g) is an equivalence class of functions, a complexity class.

Ordering of types of functions:

log(*n*) < *n* < *nK* < *kn*

**Answer (b):**

**Complexity degree**

A set A of problem has the **same complexity** under reduction r as a set B of problems if and only if A is reducible to B and B is reducible to A

(*A* =r *B*) ⇔ ((*A* r *B*) ∧ (*B* r *A*))

A complexity degree is an equivalence class under =r