STUDENT PORTOFOLIO

Name – Shivam Kumar

Reg No. – RA2011031010066

Mail: sk8589@srmist.edu.in

Department & Specialization: NWC CSE IT

SEM – 5th

Subject – 18CSC301J FORMAL LANGUAGE & AUTOMATA

FACULTY – Dr. P Balaji Srikaanth (102776)

FIA Assignment: 1 RA2011032010066 Shivam Kumar

Q. let the statement P(n) given as

P(n): $2^{2n}-1$ is divisible by 3, for every natural number n.

We observe that P(1) is true

since $2^2-1 = 4-1 = 3$

1 is divisible by 3.

Assume that P(n) is true for some natural number k

i.e, P(k): $2^{2k}-1$ is divisible by 3

$\therefore$ P, $2^{2k}-1 = 3l$. where $l \in N$

Now, to prove that P(k+1) is true

we have P(k+1): $2^{2(k+1)}-1$

$= 2^{2k+2}-1$,

$= 2^{2k} \cdot 2^2 - 1$

$= 2^{2k} \cdot 4 - 1$

$= 3 \cdot 2^{2k} + 3l = 3m$.

Thus P(k+1) is true, here even P(k) is true hence, by principle of mathematical induction natural number n.

Shivam kumar

Q.

By induction on n, As a base case is n = 5 the we have

test $5^2 = 25 < 32 = 2^5$

so, claim holds

Build inductive step, assume for $n \geq 5$ that $n^2 < 2^n$, the we have that $(n+1)^2 = n^2 + 2n + 1$

Since $n \geq 5$ we have -

$(n+1)^2 = n^2 + 2n + 1$

$< n^2 + 2n + n$ (since $1 < 5 < n$)

$= n^2 + 3n$

$(3n < 5n \leq n^2)$

$< n^2 + n^2$

$= 2n^2$

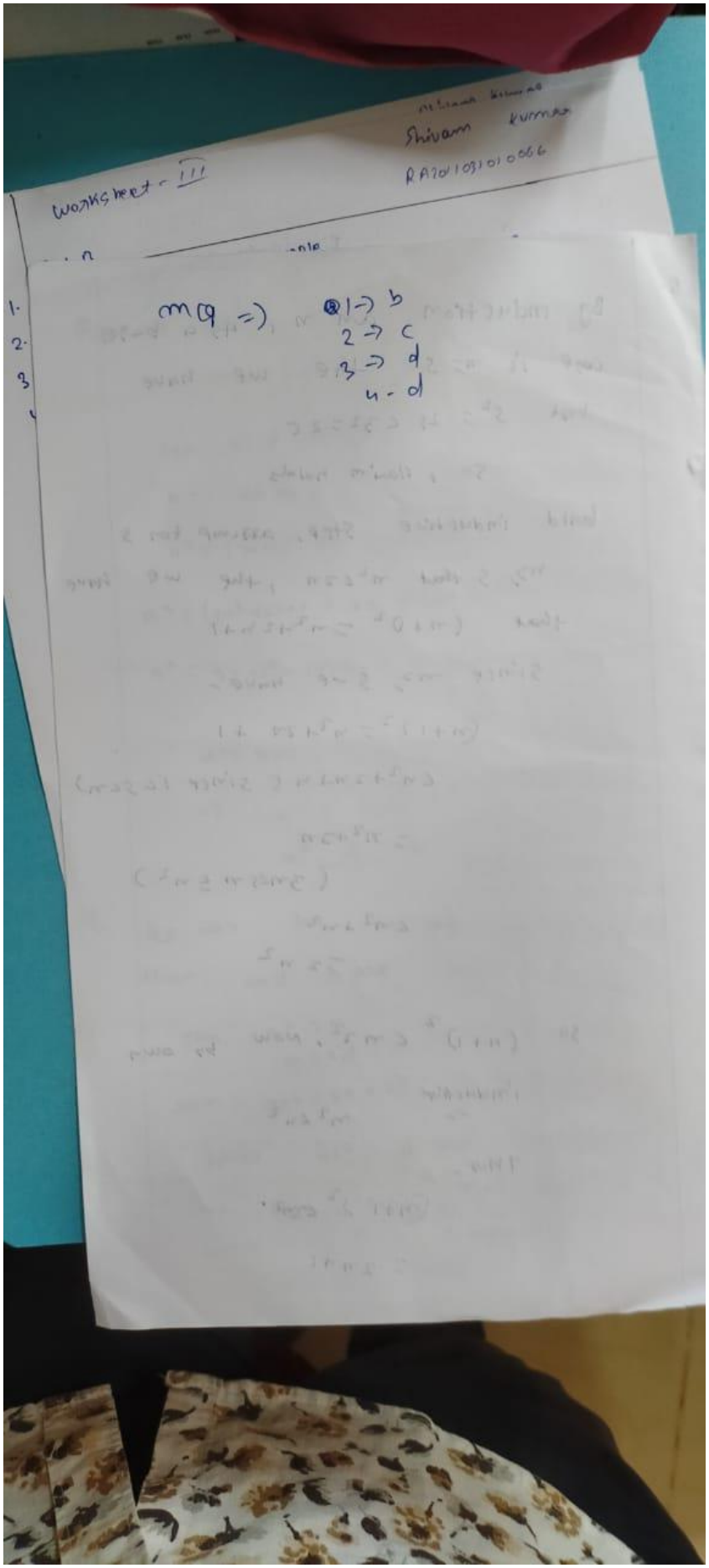
so $(n+1)^2 < n2^2$, Now by own inductive

$n^2 < n^2$

Thus -

$(n+1)^2 < 2n^2$.

$= 2n + 1$


Worksheet - III
Shivam Kumar
RA2011031010066
MCQ =>
1 -> b
2 -> c
3 -> d
4 - d

Shivam Kumar
RA2011003010066

Worksheet II

Q1. i) (a) Finite set of decimal numbers

Q2. d) logical rules

Q3. a) FSM has unique output state

Q4. b) 2

Q5. a) $(0+1)^*000(0+1)^*$

Q6. d) The recognising power is same

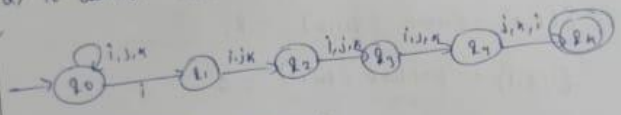
Q7. d) 4

Q8 b) move to different output state for same input

Q9 c) $(0^*10^*10^*)^*10)$

Q10 a) It cannot match second half and first half

Q11



5 tuplet structure

Q12

ϵ-closure $(q_0) = \{q_0, q_2\} \rightarrow (A)$

$\delta(A,0) = \epsilon\text{-close}(\delta(\{q_0, q_2\}), 0)$

$= \epsilon\text{-close}(\emptyset \cup q_3) = q_3 \rightarrow (B)$

$\delta(A,1) = \epsilon\text{-close}(\delta(\{q_0, q_2\}), 1)$

$= \epsilon\text{-close}(q_1 \cup q_4)$

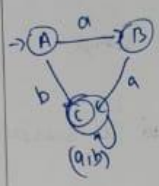
$= \{q_1, q_4\} \rightarrow C$

$\delta(B,0) = \epsilon\text{-close}(q_3, 0) = \epsilon\text{-close}(q_2)$

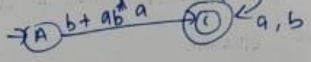
$= q_2 \rightarrow (D)$

$\delta(B,1) = \emptyset$

12



Eliminate B



$RE = (\phi + (b + ab^*a)(a+b)^*\phi)^*$

$(b+ab^*a)(a+b)^*$

$= (b+ab^*a)(a+b)^*$

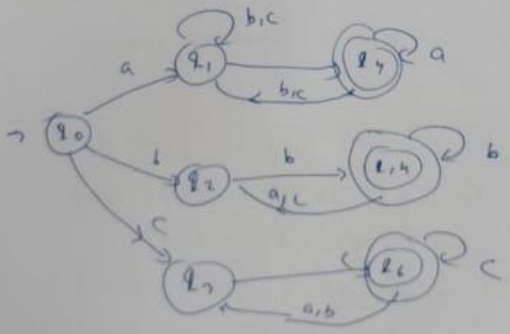
Alternatively Rij method can also be used

Worksheet - 1

Shivam Kumar

RA2011031010064

Set-D

1. a) Proof by counter example
2. b) I Denote alphabets and II denotes strings
3. b) positive closure
4. d) language that accepts strings has aaa as substring
5. b) NFA is more expensive than DFA
6. c) only ϕ indicates no transition between two states
7. c) S=T
8. d) a and ε
9. c) $Q \times \Sigma \rightarrow 2^{\wedge}Q$
10. b) only I is true
11. 

Units

1. Assume H= Head T= Tail

list R = (T, H, HTH, TT)

List S = (TH, TH, HT, T)

Now we have to find out a sequence that string formed by R and S one identical such a sequence o is 1,2,1, 3,3,4

Hence from the R and S list.

1	2	1	3	3	4	1	2	1	3	3	4
T	H	T	HTH	HTH	T	TH	TH	TH	HT	HT	T

or

take n=5

Take the combination 1,2,1,3,3,4

By concatenation strings in the series

w_1 w_2 w_1 w_3 w_3 w_4 = H_1 H_2 H_3 H_4 w_4

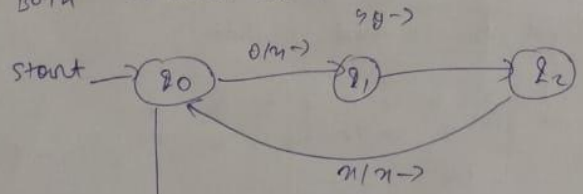
T H T HTH HTH T = TH T H HTH T HTT

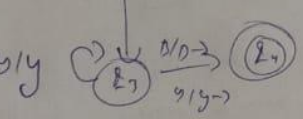
Instance ypcp = 121334

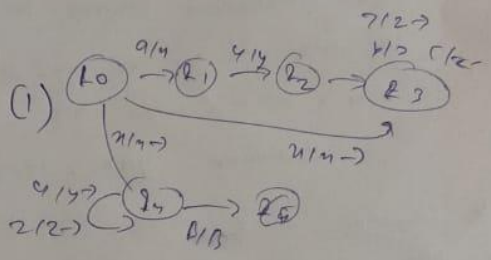
4. Counting sort takes O(n+k) time and O(n+k) space, where n is the number of items were sorting and k is the number of possible values.

we iterate through the input items twice - once to populate a way both iterate are O(n) time, Additionally we iterate through counts once to fill in next Index, which is O(k) time. The algorithm allocates 3 additional aways, one for count one for next Index and one for the output. The first two are O(k) space and the final on is (O(n)) space.

Unit-4

1. TRUE
2. TRUE
3. B) 7
4. A) left
5. A) Both S_1 and S_2 are true.
6. 



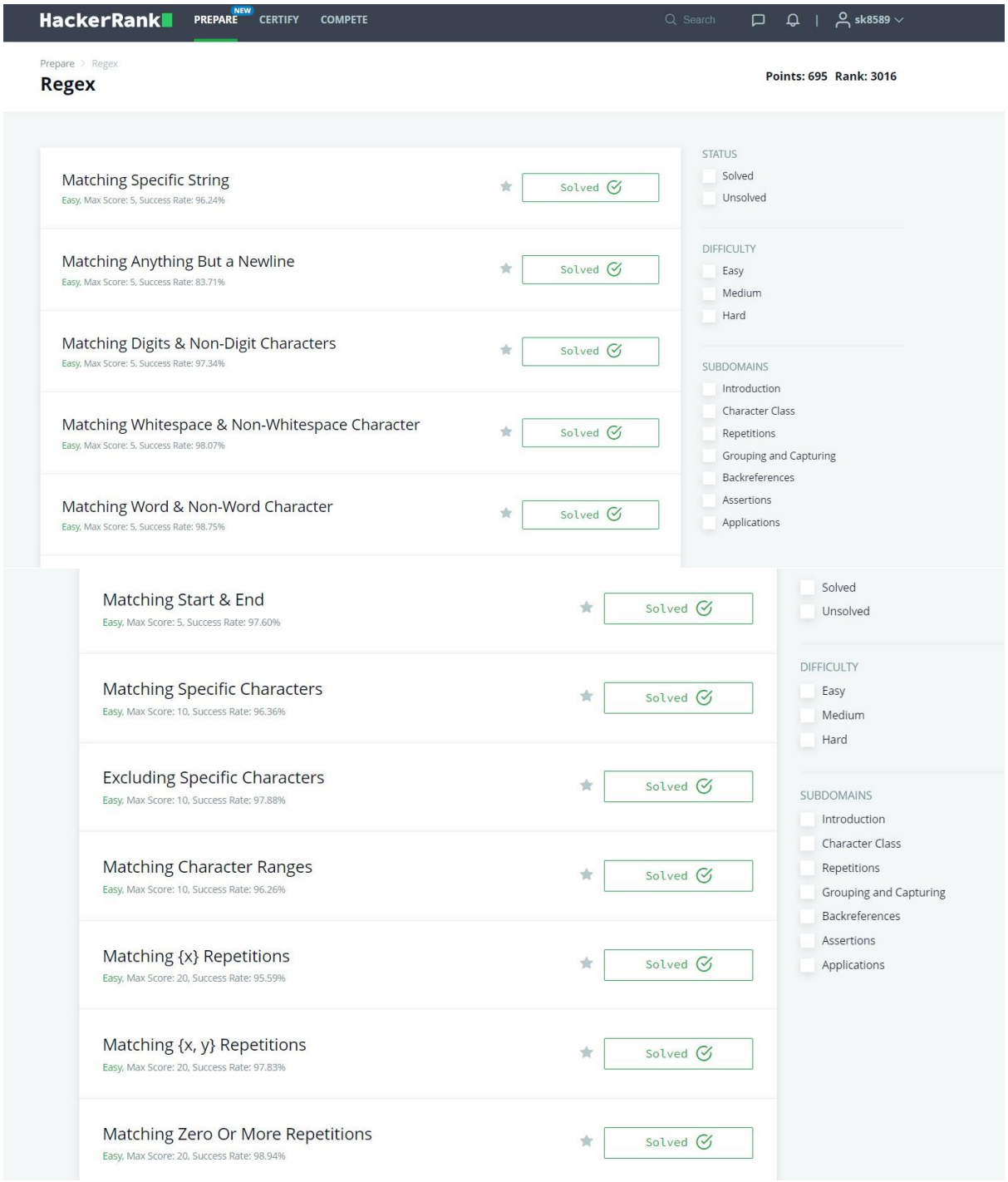
(1) 

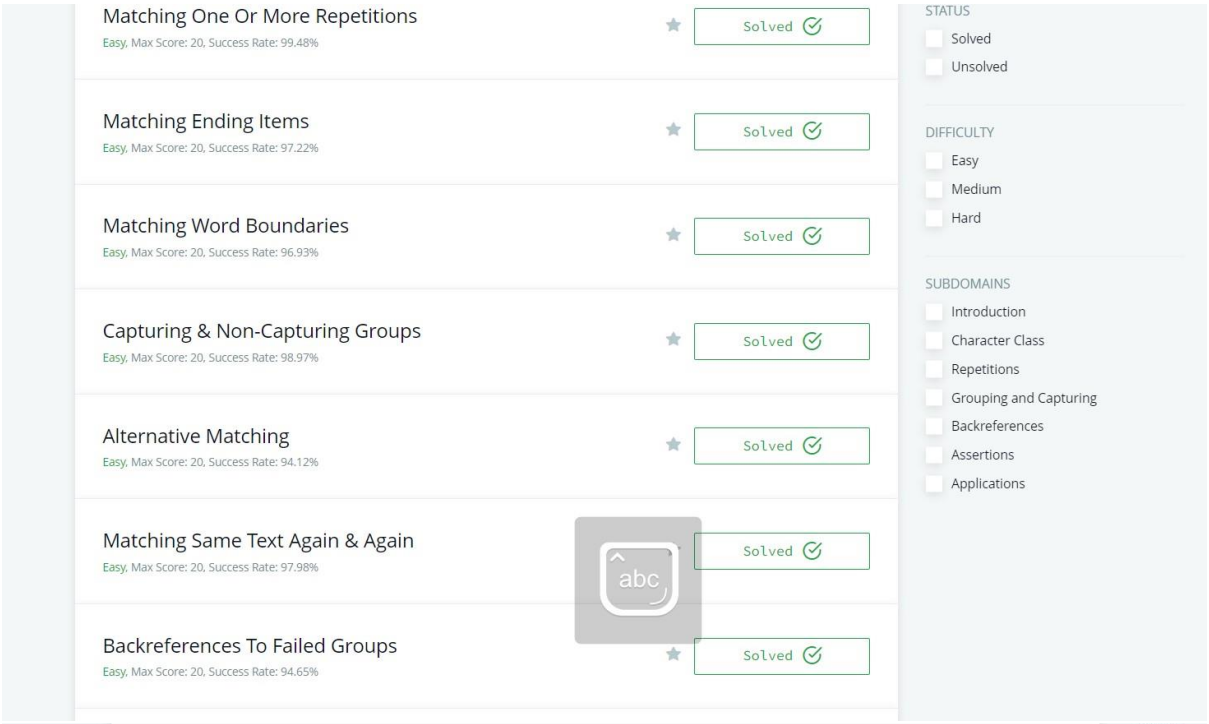
(2) multitape (multitrack) Turing machine
multi Dimensional Turing machine multidimensional tape tm

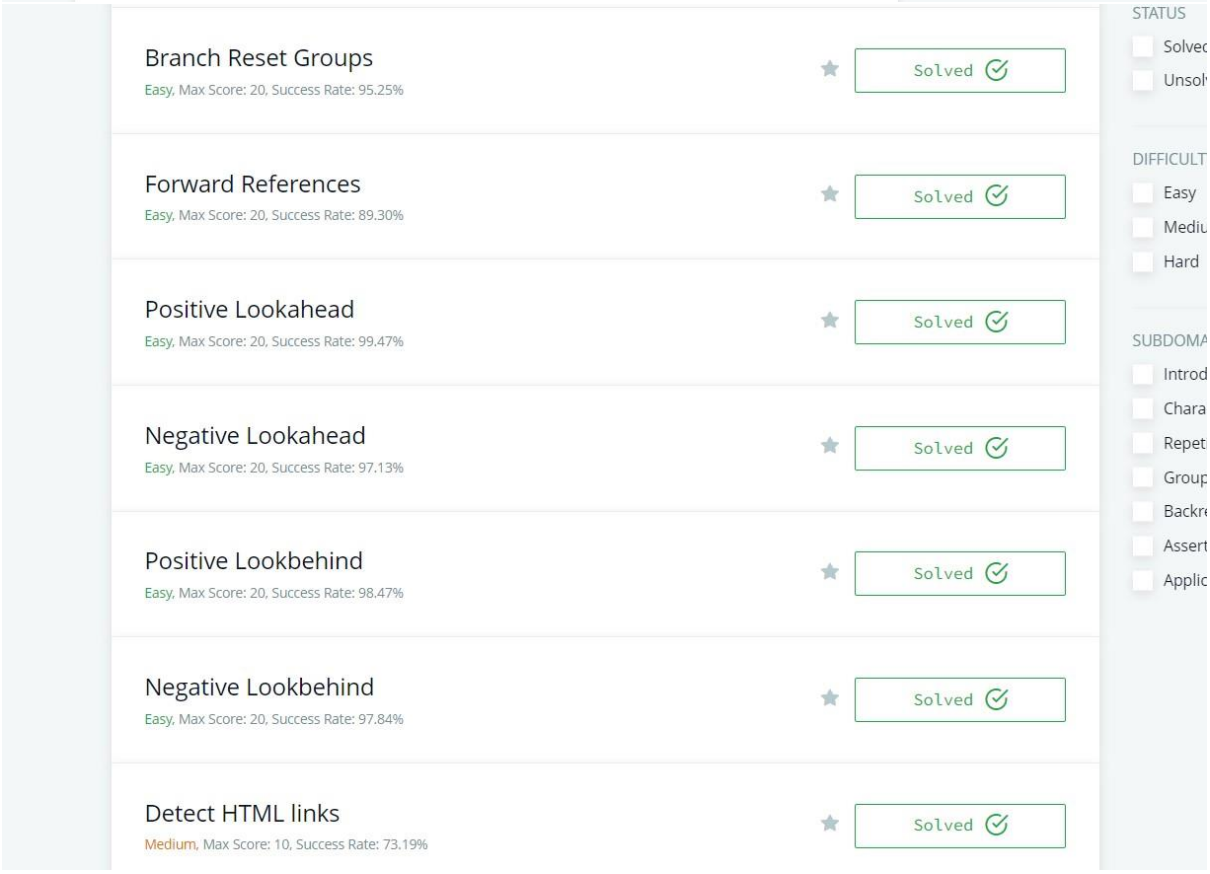
(3)

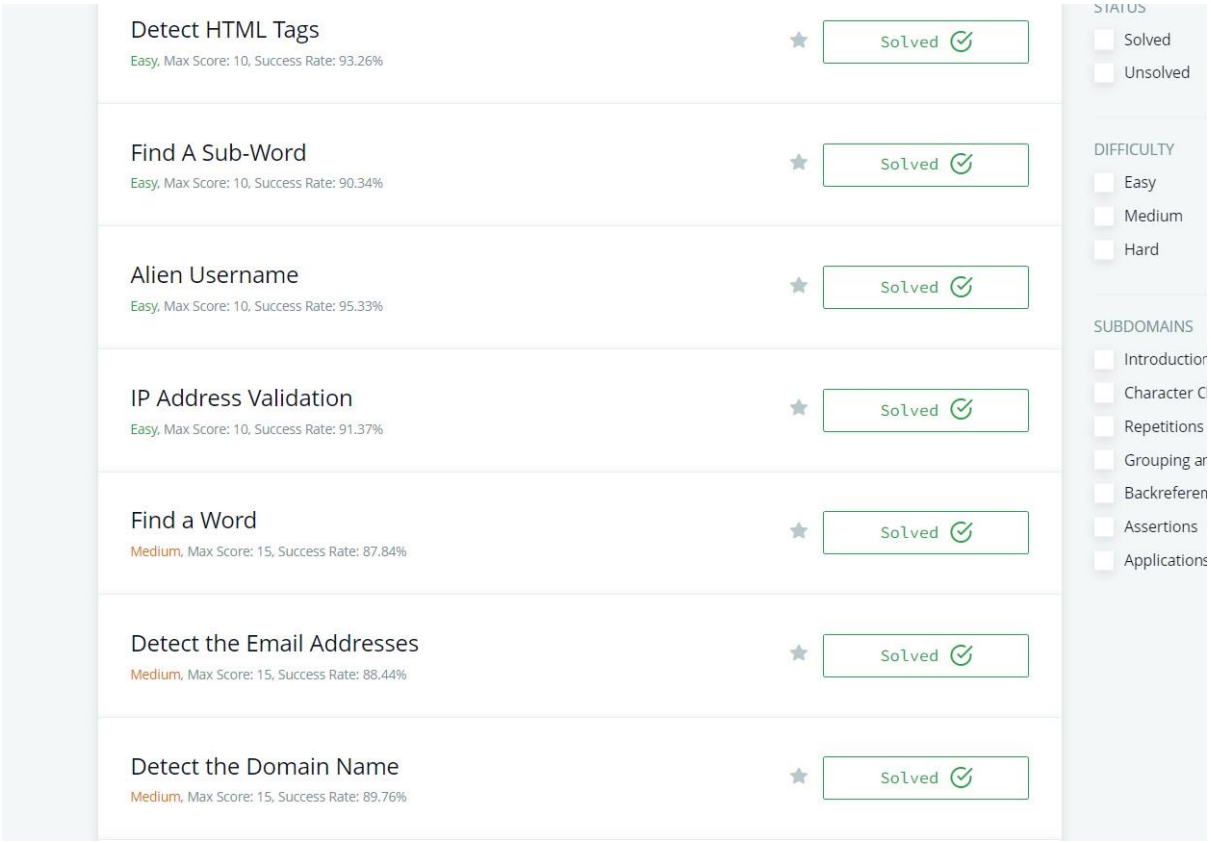
State	0	1	X	Y	B
q_0	(q_1, X, R)	-	-	(q_3, Y, R)	-
q_1	$(q_1, 0, R)$	(q_2, Y, L)	-	(q_1, Y, R)	-
q_2	$(q_2, 0, L)$	-	(q_0, X, R)	(q_2, Y, L)	-
q_3	-	-	-	(q_3, Y, R)	(q_4, B, R)
q_4	-	-	-	-	-

FLA HACKERRRANK




Matching One Or More Repetitions
Easy, Max Score: 20, Success Rate: 99.48%
Solved
Matching Ending Items
Easy, Max Score: 20, Success Rate: 97.22%
Solved
Matching Word Boundaries
Easy, Max Score: 20, Success Rate: 96.93%
Solved
Capturing & Non-Capturing Groups
Easy, Max Score: 20, Success Rate: 98.97%
Solved
Alternative Matching
Easy, Max Score: 20, Success Rate: 94.12%
Solved
Matching Same Text Again & Again
Easy, Max Score: 20, Success Rate: 97.98%
Solved
Backreferences To Failed Groups
Easy, Max Score: 20, Success Rate: 94.65%
Solved
STATUS
Solved
Unsolved
DIFFICULTY
Easy
Medium
Hard
SUBDOMAINS
Introduction
Character Class
Repetitions
Grouping and Capturing
Backreferences
Assertions
Applications


Branch Reset Groups
Easy, Max Score: 20, Success Rate: 95.25%
Solved
Forward References
Easy, Max Score: 20, Success Rate: 89.30%
Solved
Positive Lookahead
Easy, Max Score: 20, Success Rate: 99.47%
Solved
Negative Lookahead
Easy, Max Score: 20, Success Rate: 97.13%
Solved
Positive Lookbehind
Easy, Max Score: 20, Success Rate: 98.47%
Solved
Negative Lookbehind
Easy, Max Score: 20, Success Rate: 97.84%
Solved
Detect HTML links
Medium, Max Score: 10, Success Rate: 73.19%
Solved
STATUS
DIFFICULT
Easy
SUBDOMA


Detect HTML Tags
Easy, Max Score: 10, Success Rate: 93.26%
Solved
Find A Sub-Word
Easy, Max Score: 10, Success Rate: 90.34%
Solved
Alien Username
Easy, Max Score: 10, Success Rate: 95.33%
Solved
IP Address Validation
Easy, Max Score: 10, Success Rate: 91.37%
Solved
Find a Word
Medium, Max Score: 15, Success Rate: 87.84%
Solved
Detect the Email Addresses
Medium, Max Score: 15, Success Rate: 88.44%
Solved
Detect the Domain Name
Medium, Max Score: 15, Success Rate: 89.76%
Solved
STATUS
Solved
Unsolved
DIFFICULTY
Easy
Medium
Hard
SUBDOMAINS
Introductior
Character C
Repetitions
Grouping ar
Backreferen
Assertions
Applications