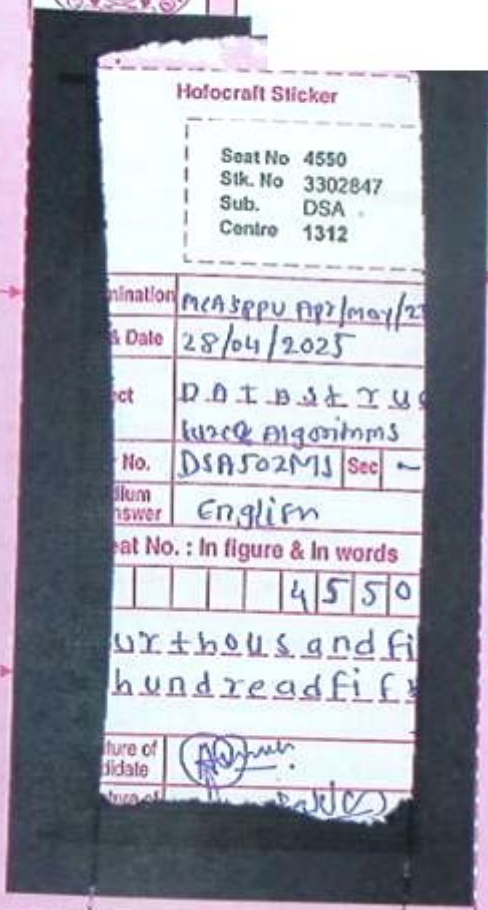


Hofocraft Sticker

Seat No 4550
Stk. No 3302847
Sub. DSA
Centre 1312

Examination	MCASEPPU Apr/May/2
Date	28/04/2025
Subject	DATA STRU Work Algorithms
Paper No.	DSA502MJ Sec
Medium of Answer	English
Seat No. : In figure & In words	4550 Four thousand fi hundred fifty
Signature of Candidate	
Signature of Invigilator	

SAVITRIBAI PHULE PUNE UNIVERSITY

DSA502MJ-DSA

Sem:1 20324

Instruction to Candidate

1. Candidate has to confirm seat number, subject and centre number printed on Bar code and Write it on attendance sheet.
   विद्यार्थ्यांनि प्रथम बार कोडवरील आसन क्रमांक, विषय व केंद्र क्रमांक तपासून घेण्य उपस्थिती पत्रकावर नोंदवावी.
2. Paste Bar Code in prescribed space.
   उत्तरपत्रिकेवरील लिहित जागेतच बार कोड लावावा.
3. Do not write anything on Bar code sticker, otherwise it will be treated as unfairmeans.
   बार कोड स्टिकरवर काहीही लिहू नये, अन्यथा परीक्षा गैरमार्ग समजला जाईल.

Q. No.	Examiner	Moderator
1	0 0	
2	0 2	
3	0 4	
4	0 5	
5	0 2	
6		
7		
8		
9		
10		
11		
12		
Total in Figure	0 1 3	
Total in Words	Thirteen	
Signature		

Specific remarks regarding malpractice (in Red Ink)

Total	Marks in Figure	Marks in Words	Sign
Examiner	13	Thirteen	
Moderator			

१. विद्यार्थ्याने उत्तरपत्रिके च्या मुखपृष्ठावर तसेच उपस्थिती पत्रकावर विहित जागेत आसन क्रमांक अंकात व अक्षरात बिनचूक लिहून स्वाक्षरी करावी.

२. उत्तरपत्रिकेवर फक्त निळ्या अथवा काळ्या शाईचा उपयोग करावा, अन्यथा उत्तरपत्रिकेचे मूल्यमापन केले जाणार नाही.

३. उत्तरपत्रिकेतल्या पृष्ठक्रमांक ३ पासून लिहिण्यास प्रारंभ करावा.

४. संबंधित प्रश्नाचे अथवा उपप्रश्नाचे उत्तर जेथून सुरू होते तेथेच समासात प्रश्न क्रमांक, उपप्रश्न क्रमांक अचूक व स्पष्ट लिहावा, यासाठी वेगळ्या शाईचा उपयोग करू नये.

५. प्रत्येक पानाच्या दोन्ही बाजूस लिहावे, उत्तरपत्रिका किंवा पुरवणी उत्तरपत्रिकेचे कोणतेही पान फाडू नये, फाडल्यास परीक्षा गैरप्रकार समजून पुढील कार्यवाही करण्यात येईल.

६. पेपर संपण्यापूर्वी १० मिनिटे अगोदर इशारा घंटा होईल, त्यानंतर विद्यार्थ्याने उत्तरपत्रिका व पुरवणी उत्तरपत्रिकेवर होलोक्राफ्ट स्टिकर विहित जागेवरच लावावा.

७. कॉपी करणे किंवा दुसऱ्याच्या नावावर परीक्षेस बसणे यांसारख्या कृती 'महाराष्ट्र-प्रीव्हेन्शन ऑफमाल्प्रॅक्टिस ॲक्ट युनिव्हर्सिटी, बोर्ड ॲण्ड अदर स्पेसिफाईड एक्झामिनेशन्स ॲक्ट, १९८२' (सा.फु.पु.वि. चा अध्यादेश क्रमांक १) त्यानुसार संमत केलेला कायदा या अन्वये दंडही असेल.

Candidate shall fill all information about seat number, paper etc. in prescribed space and sign on the answer book and attendance sheet.

Candidate shall use blue or black ink only. Otherwise answer book will not be evaluated.

Candidate shall start writing answer from page no. 3 of the answer book.

Candidate shall mention question number, sub question number correctly at the beginning of the same and shall not use ink other than blue or black.

Candidate shall write on both sides of pages and shall not tear off any page, it will be treated as unfairmeans.

Warning bell will be given before 10 minutes of the concluding time. Candidate shall paste Holocraft Sticker at appropriate space on the answer book.

An Act of Copying or Impersonations at an Examination is Punishable under 'The Maharashtra Prevention of Malpractices at University, Board and Other Specified Examinations Act, 1982'(Ordinance 9 of SPPU). The Act passed to the effect.

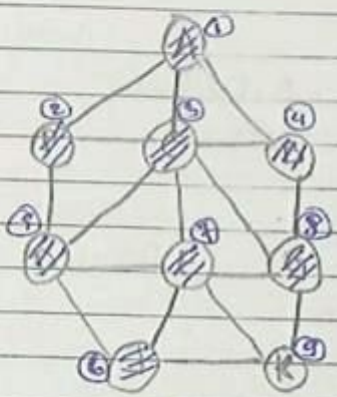
Examiner and Moderator has to write marks on all given appropriate place only. Examiner should give assessment tick(✓) or (×) in the margin.

Q. No.	Examiner	Moderator	Verification	Revaluation
1	00			
2	02			
3	04			
4	05			
5	02			
6	|			
7	|			
8	|			
9	|			
10	|			
11	|			
12	\			
Total	13			

Q. No. /Q.No.

Q. 4-a

Ans -



Ans :- Depth first search
we have to perform a algorithm for depth search.

Step-1 initialize
initialize the one pointer from given graph - as [A]
from 'A' we perform.

iteration - 1
Enqueu = A Dequeu = A
Adjacency of A = f, c, B
visited = A
Dequeu = A

iteration - 2
Enqueu = f Dequeu = f
Adjacency of f = A, c
visited = A, f
Dequeu = f

iterator - 3
inqueue = C Dequeue = C
Adjuncy of C = A, F, B, D, G, E
visited = A, F, C
Deque = C

iterator - 4
inqueue = B Dequeue = B
Adjuncy of B = A, G, C
visited = A, F, C, B
Deque = B

iterator - 5
inqueue = D Dequeue = D
Adjuncy of D = F, J, E, C
visited = A, F, C, B, D
Deque = D

iterator - 6
inqueue = J Deque = J
Adjuncy of J = D, E, K
visited = A, F, C, B, D, J
Dequeue = J

iterator - 7
inque = E Deque = E
Adjuncy of E = C, D, G, J, K
visited = A, F, C, B, D, J, E
pequeue = E

iterator - 8
inqueue = G
Adjancy of G = B, C, E, K. Dequeu = G
visited = A, F, C, B, D, J, E, G
Deque = G

iterator - 9
inqueue = K
Adjancy of K = G, E, I Dequeu = K.
visited "K"
remaining is already visited.
Deque = K

(3) as all the ports are visited and print the result

So (A)->(F)->(C)->(B)->(D)->(J)->(E)->(G)->(K)
the full pruned DFS after applying algorithm step by step
that is the final output.

	Q.No.	3(c)					TOTAL
E		2					2
M							

Y. M. /Q.No.

Q.3 C

– A*(B*(C/(D*E)-F))+G

Symbol	stack	postfix
A		A
*	*	A
(	*(	A
B	*(	AB
*	*(*	AB
(	*(*(	AB
C	*(*(	ABC
/	*(*(/	ABC
(	*(*(/(	ABC
D	*(*(/(	ABCD
*	*(*(/(*	ABCD
E	*(*(/(*	ABCDE
)	*(*	ABCDE*(/(
–	*(–	ABCDE*(/(**
F	*(–	ABCDE*(/(** F
)	*(–)	ABCDE*(/(** F
)	*(–))	ABCDE*(/(** F
+	*(– +	ABCDE*(/(** F
G	*(– +	ABCDE*(/(** FG – +

postfix exp :- ABCDE*(/(**FG-+

Q. /Q.No.

Q.5

a

=> 25, 83, 4, 2, 75, 16, 45, 5, 99, 1

Firstly sort the array in ascending order without ascending order we could not find any algorithm.

	0	1	2	3	4	5	6	7	8	9
arr =	1	2	4	5	16	25	45	75	83	99

So as we don't have any search key in question so we can assume any key from the following array

Search key = [75]

we took 75 as search key.

Step1 = initialize

~~mid = 0~~

Low = 0

high = 9

and for finding mid the formula is

~~mid~~ $\frac{Low + high}{2}$

Step2 - iteration - 1

mid = (0+9)//2 = 4

arr[mid] = 16

16 is less than 75 -> search right half only

update low = mid + 1 = 5

	Q.No.						TOTAL
E							
M							

S. E. /Q.No.

Iteration - 2

mid = (5+9) // 2 = 7

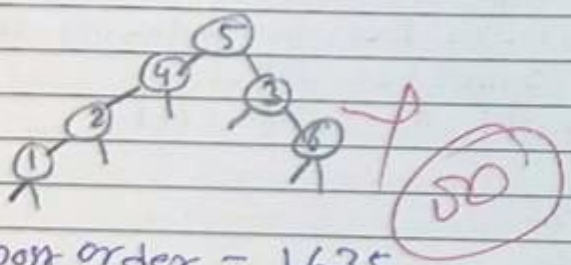
arr[mid] = 75

75 = 75 search [redacted].

∴ Search key = 75 array[7] = 75

& - found the element at index = 7.

Q.4(b)



post order = 1, 6, 3, 5

Q.No.

Q.2

b

Singly linked list and circular linked list is used store data in allocation by node format in DSA it's help for search and sort the data..

and save time, without tenory process and also manage the memory allocation management.

it's helpful for user's to manage the data in the linked list format

in DSA we have 3 types of linked list.

although somewhere circle list do same work, but the work flow work style of every linked list is different.

we can see the difference between singly and circular linked list.

Aspect	singly linked list	circular linked list
insertion of Node	in singly linked list insertion of node is easy cause there is only forward moving	insertion of node is difficult as node can add in both forward and previous side.

	Q.No.						TOTAL
E		2(b)					2
M		2					

Q. K./Q.No.

	Singly linked list	circular
Aspect		
Creation of Node	Creation of node is also easy Because only one side forward of linked list we can create	compare to singly creation of node cause we have to lookup both side of the linked list.
time consuming	singly linked list is less time consumed than circular linked list	compare to singly linked list. consume more time.
memory allocation	easy memory allocation than circular linked list	at this side memory allocation is hard because of prev and next node.
using method	easy to use because of singly linked list only move to the forward side	hard to use because of both side can move previous and forward.
Application	queue, booking. etc.	navigation, map, music player.

प्र. क्र./Q.No.

Q5
b

Hash function in Data Structure and algorithms · Stands for.
in hash table we can manage the data and value to the index or any table.

Hashing table is table called hash table.
where we can add data and remove or perform operation as we need.

With the help of hashin as we know if we want to to add or remove data at any ~~data~~ table or index value we have to do a managing tempory or give iterator.

traverse function every node and the after finding the node we can insert the data, or remove.
With the help of hash table we can mention directly index position and directly at this position we can add remove data

without wasting time.

	Q.No.						TOTAL
E							
M							

Let say
Ex. 25, 15, 5 – index = 5

So assume three element are trying to insert in the index position of =5 five.

① 25 → element
mod → index - 5

② 15 → element
mod → index 5
collision - index - 5

③ 5 → element
mod → index 5
collision → index - 5

now here because of three element is trying to occur in same index position.

So here collision is occuring for managing the collision

we have chain system or blockwise system.
but the best method is chaining method this can manage combine the elements at the one place and chaining them.

like

[25 ↔ 15 ↔ 5 ↔]

and now after chaining the elements they can be stored in

[25 ↔ 15 ↔ 5 ↔]

now we can store this as in index position of =[5].
that's the hashing help to manage.

Q.3
d-

a queue is used for store data in a formative manner's style.
queue use Fifo [first in first out]
enviornment.

queue has various application and operation to perform on data or storage management.

linear :-
manages data in linear format line by line.
systematically so no errors occurs happens.

linear queue is like :
real life example :- Booking theater ticket

	Q.No.					TOTAL
E		2				2
M						

T. F. /Q.No.

for movies the one who came first in queue will be served first.

Operations:-
Linear :- insert data or delete data on line-by-line method

inQueue :- insert data into queue from the manner Push method

deQueue :- delete data from queue from it's following method.

peek :- find the element as if need any particular position

isempty :- print the result and return the element from queue

circular :- insert and find data from both side peri' rear and front side.
this can manage from both side.

Applications :- network management, operating system, Database management system etc.

S./Q.No.

Q 2

a

1 In between of the list

Step-1 Start

Step-2 -

Check if the linked list is empty or not `Head == null`

if empty
print "list is empty"

Step-3 Create a new node. and assign to the Head node new node = Head node.

Step-4 Create a new node for adding between in the list.
current.next.next = node.null

now traverse the current step upto the finding position

Step-5 reache at the between of list now insert the element at this position

Step-6 exit

S. E./Q.No.

ii) At the end of the list

Step-1 Start

Step-2 check if the linked list is empty or not.
if list is empty
print "List is empty"

Step-3 assign a new node to the head
new.node == Head.node.

Step-4 now create a temprary iteratorator for traversing
i = current.next.next.next = null
add the element

Step-5 now traverse teh temp and once reach the perticular insert the element.

Q. /Q.No.

Q.7

b

String manipulation using arrays for finding the element or changing the element or data set at position managing string manipulation.

In array format array can store data in inbuilt ascending format with memory execution. and

from array we can perform various application, like.

In array forming, inserting, deleting and printing the array result we have many more operation like this in DSA.

and that's all gone through the array list for this operation such data we have the string manipulation method in data structure algorithms.

From array we can manipulate the string easily while attempting any operation or the assertion

string manipulation can help to perform operation non texah. like its going to hard.

	Q.No.					TOTAL
E	3(b)					10
M						

Q. No./Q.No.

we can create the string and manipulate them as we need or user want to do while use in operation.

In Data structure and algorithms array list can provide multiple work anyone for storing true elements and allocation of memory.

and for such type we have perform various operation on array list for that kind of stuff we can took help from strings

that can manage every thing to manipulation of strings, in array list for DSA, and it's give full work standard way.

instead of hard to use and wasting too many time to manage the operation that's why string manipulation is one of the most important function in array list.

Q. /Q.No.

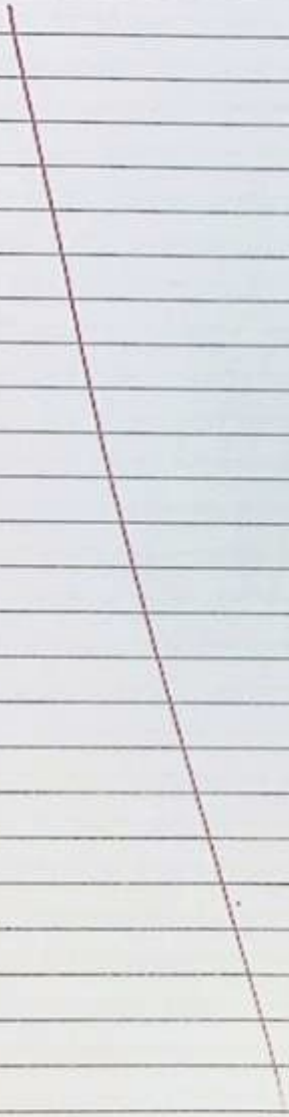
Q.Q1

	Q.No.						TOTAL
E							
M							

S. R. /Q.No.

	Q.No.						TOTAL
E							
M							

प्र. क्र. /Q.No.



Q. /Q.No.

	Q.No.						TOTAL
E							
M							

Q. No. /Q.No.

Rough Work

pre - Root Left Right
in Right Root Left
post Left Right Root

in-order - 4, 2, 5, 1, 3, 6

pre-order - 1, 2, 4, 5, 3, 6

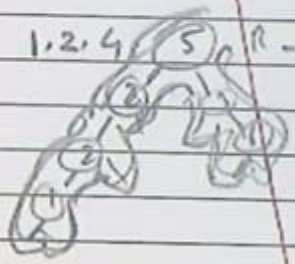
1, 2, 4 (5) Ri 3, 6

A*(B*(C/D*E))

$\frac{0+9}{2}$ $\frac{9}{4}$

5+9 $\frac{14}{2}$ 7.

1, 2, 4 (5) R - 3 6



4 2 5 (1) 3 6

5, 2, 1, 3, 6

1 2 4 5 3 6

1 6 3 5