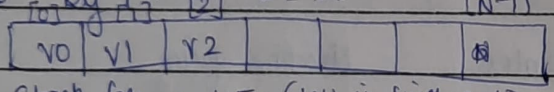
DATE: / /
PAGE NO.:

Review sheet

Q1) (a) → Reverse a string → Holding all data of a function / recursion → Memory management → Evaluating expression → Parsing
give example

(b) The use of 1-d array as a static stack means that the stack can become full. There are 2 ways of doing this
① To reserve the first element of the array for stack pointer, which would contain the index of the current top element in the stack. If the stack pointer is stored in array[0] then its initial value will be zero. If an element is pushed now, the stack pointer value increases to 1 and new element is stored in array[1]. A[TOP] = VALUE
② If the element is popped from stack, the top element in the array will be set to null and the pointer value will ↓ by 1.



Stack follows LIFO (last in first out)

isEmpty() → reports if stack is empty or not and returns true if TOP < 0.

isFull() → Returns true if stack is full. TOP > N-1 so true return.

Q2) (a) 4

(b)
```
SET = False
loop for I from 0 to N-1
    loop for J from 0 to N-1
        if I = J then
            if MAT[I,J] != 0 then
                SET = True
                I = 0
                J = 0
            end if
        end if
        if MAT[I+1,J] != 0 AND MAT[I,J+1] != 0 then
            SET = True
            MAT[I+1,J] = 0
            MAT[I,J+1] = 0
        end if
```

DATE: / /
PAGE NO.:

```
        if MAT[I,J] = 0 then
            SET = True
        end if
    end loop
end loop
output SET
```

Q2 (c) mystery (MAT, 5)

↓

MAT[5][4] + mystery (MAT, 4)

↓

MAT[4][3] + mystery (MAT, 3)

↓

MAT[3][2] + mystery (MAT, 2)

↓

MAT[2][1] + mystery (MAT, 1)

↓

MAT[1][0] + mystery (MAT, 0)

↓

0

$5+7+(-5)+9+1$

$= \boxed{17}$

(d) It finds the sum of elements in the lower subdiagonal.

Q3 (b)

```
INVALID = False
R=0
loop while R<N and not INVALID
    C=0
    loop while C<N and not INVALID
        if abs(R-C) >= 2 and A[R][C] != 0 or abs(R-C) <2 then
    and A[R][C] == 0 then
            INVALID = True
        end if
        C=C+1
end while
```

R = R+1

end while

return not INVALID