

**NATIONAL UNIVERSITY OF SINGAPORE**  
Department of Computer Science, School of Computing  
**IT5001—Software Development Fundamentals**  
Academic Year 2023/2024, Semester 2  
**MID-TERM ASSESSMENT**  
Solutions Manual

17 February 2024

**Time allowed:** 1 hour

---

## **EXPRESSION EVALUATION [20 marks]**

Evaluate the following expressions.

**Question 1** [2 marks]. `1 / 2 - 3 / 4`. *Answer:* `-0.25`

**Question 2** [2 marks]. `'abc' * 2`. *Answer:* `'abcabc'`.

**Question 3** [2 marks]. `(1, 2, 3) + (4,)`. *Answer:* `(1, 2, 3, 4)`.

**Question 4** [2 marks]. `int(-2.5)`. *Answer:* `-2`.

**Question 5** [2 marks]. `[1, 2] + ([3, 4])`. *Answer:* `[1, 2, 3, 4]`. `(([3, 4]))` is just a list in parentheses

**Question 6** [2 marks]. `['IT'].append(5001)`. *Answer:* `None`. (append returns `None` and produces the side-effect of appending into the list).

**Question 7** [2 marks]. `[(1), [2)][0]`. *Answer:* `1`.

**Question 8** [2 marks]. `not 1 or else 2`. *Answer:* `SyntaxError`.

**Question 9** [2 marks]. `[4, range(5)][[1, 0][0]:][0][2]`. *Answer:* `2`.

**Question 10** [2 marks]. `[i * 2 for j in range(2) for i in range(2) if j]`. *Answer:* `[0, 2]`.

## **TRUE/FALSE QUESTIONS [10 marks]**

**Question 11** [2 marks]. The expression `1 / 0` raises an exception because we cannot divide by zero.  
*Answer:* True.

**Question 12** [2 marks]. If `ls` is a list with three elements, then `range(len(ls))` produces the sequence of integers 0, 1 and 2.  
*Answer:* True.

**Question 13** [2 marks]. Assuming `ls` is a list, the expression `ls and ls[0]` raises an `IndexError` if `ls` is empty. *Answer:* False (short-circuit logic).

**Question 14** [2 marks]. The expression `[(i, i + 1) for i in 'abcd']` produces a list with four elements. *Answer:* False (error raised on `i + 1`).

**Question 15** [2 marks]. Assuming `ls` is a list, the statement `ls += (1, 2)` causes the list object referred to by `ls` to be mutated. *Answer:* True (`+=` on lists implicitly extends).

## PROGRAM TRACING [20 marks]

**Question 16** [6 marks]. Determine the output of running the following program:

```
def f(s):
    if not s:
        return ''
    return f(s[2:]) + s[0]
print(f('abcde'))
```

*Answer:* eca.

**Question 17** [7 marks]. Determine the output of running the following program:

```
def g(ls):
    res = []
    for i in ls:
        if not isinstance(i, list):
            res.append(i)
        else:
            for j in i:
                res.append(j)
    return res
print(g([[1, 2], 3, [4, 5]]))
```

*Answer:* [1, 2, 3, 4, 5].

**Question 18** [7 marks]. Determine the output of running the following program:

```
def h(ls):
    res = []
    t = 0
    for i in ls:
        t += i
        res.append(t)
```

```
    return res
print(h([1, 2, 3, 4, 5]))
```

*Answer:* [1, 3, 6, 10, 15].

## FILL IN THE BLANKS [50 marks]

**Question 19** [10 marks]. Complete the function `digit_swap(n)` that takes in an integer `n` between 11 and 99 inclusively and return an integer that the digits are swapped. *Answer:*

```
def digit_swap(n):
    return n % 10 * 10 + n // 10
```

**Question 20** [10 marks]. Complete the function `diff_strings(s1, s2)` that takes in two strings of equal length, and return the number of characters with the same index in the two strings that are different. Do so iteratively. *Answer:*

```
def diff_strings(s1, s2):
    count = 0
    for i in range(len(s1)):
        if s1[i] != s2[i]:
            count += 1
    return count
```

**Question 21** [10 marks]. Complete the function `diff_strings(s1, s2)` that takes in two strings of equal length, and return the number of characters with the same index in the two strings that are different. Do so recursively. *Answer:*

```
def diff_strings(s1, s2):
    if not s1:
        return 0
    if s1[0] == s2[0]:
        return diff_strings(s1[1:], s2[1:])
    else:
        return 1 + diff_strings(s1[1:], s2[1:])
```

**Question 22** [10 marks]. Complete the function `even_index_list(lst)` that takes a list `lst` and returns another list with only the items with even indices in `lst`. Do so iteratively. *Answer:*

```
def even_index_list(lst):
    output = []
```

```
for i in range(0, len(lst), 2):
    output.append(lst[i])
return output
```

**Question 23** [10 marks]. Complete the function even\_index\_list(lst) that takes a list lst and returns another list with only the items with even indices in lst. Do so recursively. *Answer:*

```
def even_index_list(lst):
    if not lst:
        return []
    return [lst[0]] + even_index_list(lst[2:])
```

– End of Solutions Manual –