

Quiz 1

Deadline	Friday, 25 September 2020 at 5:00PM
Latest Submission	<i>no submission yet</i>
Maximum Mark	4

Question 1 (1 mark)

What is the big-O characterisation in terms of n , of the running time of the following code segment?

```

Algorithm q1(A, B):
    Input: arrays A and B each containing  $n \geq 1$  integers (they both contain the same r
    Output: <unknown>

    int c = 0;
    for (int i = 0; i < n; i++) {
        int s = 0;
        for (int j = 0; j < n; j++) {
            s = s + A[0];
            for (int k = 1; k < j; k++) {
                s = s + A[k];
            }
        }
        if (B[i] == s) {
            c++;
        }
    }
    return c;

```

Note: (x^y) means x^y .

(a) <input type="radio"/>	$O(1)$
(b) <input type="radio"/>	$O(\log n)$
(c) <input type="radio"/>	$O(n)$
(d) <input type="radio"/>	$O(n \log n)$
(e) <input type="radio"/>	$O(n^2)$
(f) <input checked="" type="radio"/>	$O(n^3)$

(g) <input type="radio"/>	$O(2^n)$
---------------------------	----------

Question 2 (1 mark)

What is the big-O characterisation in terms of n , of the running time of the following code segment?

```

Algorithm q2(A):
  Input: array A containing  $n \geq 1$  integers
  Output: <unknown>

  int select_sum = 0;
  int i = n - 1;
  while (i > 0) {
    select_sum = select_sum + A[i];
    i = i / 2;
  }
  return select_sum;

```

Note: (x^y) means x^y .

(a) <input type="radio"/>	$O(1)$
(b) <input checked="" type="radio"/>	$O(\log n)$
(c) <input type="radio"/>	$O(n)$
(d) <input type="radio"/>	$O(n \log n)$
(e) <input type="radio"/>	$O(n^2)$
(f) <input type="radio"/>	$O(n^3)$
(g) <input type="radio"/>	$O(2^n)$

Question 3 (1 mark)

What is the big-O characterisation in terms of n ($n = hi - lo$), of the running time of the following code segment?

```
Algorithm q3(A, lo, hi):
    Input: array A containing m integers,  $0 \leq lo \leq hi < m$ 
    Output: <unknown>

    if (lo >= hi)
        return;

    int value = (A[lo] + A[hi]) / 2;
    if (value > 10) {
        printf("%d %d", lo, hi);
        return;
    }
    q3(A, lo + 1, hi);
```

Note: (x^y) means x^y .

(a) <input type="radio"/>	$O(1)$
(b) <input type="radio"/>	$O(\log n)$
(c) <input checked="" type="radio"/>	$O(n)$
(d) <input type="radio"/>	$O(n \log n)$
(e) <input type="radio"/>	$O(n^2)$
(f) <input type="radio"/>	$O(n^3)$
(g) <input type="radio"/>	$O(2^n)$

Question 4 (1 mark)

What is the big-O characterisation in terms of n ($n = hi - lo$), of the running time of the following code segment?

Algorithm q4(A, lo, hi):

Input: array A containing m integers, $0 \leq lo \leq hi < m$

Output: <unknown>

```
if (lo >= hi)
    return;

int j = (lo + hi) / 2;
if (A[lo] < A[hi]) {
    q4(A, lo, j);
    q4(A, j, hi);
}
else {
    q4(A, j, hi);
    q4(A, lo, j);
}
for (int i = lo; i < hi; i++) {
    print("%d", A[i]);
}
```

Note: (x^y) means x^y .

(a) <input type="radio"/>	$O(1)$
(b) <input type="radio"/>	$O(\log n)$
(c) <input type="radio"/>	$O(n)$
(d) <input type="radio"/>	$O(n \log n)$
(e) <input type="radio"/>	$O(n^2)$
(f) <input type="radio"/>	$O(n^3)$
(g) <input checked="" type="radio"/>	$O(2^n)$

✓ Submit