

Topic 3 · Sheet A: Questions

Analysing the runtime of an algorithm

Suggested time

About 75 minutes

Total marks

73

Questions

12

Builds on

Topics 1 to 2

How to use this sheet. Work on your own paper. The questions start gently and get harder; questions 11 and 12 are exam style. Marks are a guide to how much working is expected, roughly one minute per mark. You need to count how often each step of an algorithm runs, and the worst-case asymptotic running time $t(n) = \max$ over inputs of size n of the time taken.

Notation. Arrays are $A[1, \dots, n]$. A **return** ends the algorithm at once. “Comparisons” means executions of the test in an **if** line.

1

4 marks

Algorithm ZEROPAIR decides whether two distinct entries of A add up to 0.

```
1 for  $i = 1, \dots, n$  do
2   for  $j = i + 1, \dots, n$  do
3     if  $A[i] + A[j] = 0$  then return YES
4 return NO
```

How many comparisons does it make on each input?

- (a) $A = [3, -3, 5, 1]$ [1]
- (b) $A = [1, 2, 3, 4]$ [1]
- (c) $A = [2, 5, -5, 7]$ [1]
- (d) Any array of size n with no pair adding to 0 [1]

2

4 marks

Algorithm MAX finds the largest entry of A .

```
1  $m \leftarrow A[1]$ 
2 for  $i = 2, \dots, n$  do
3   if  $A[i] > m$  then  $m \leftarrow A[i]$ 
4 return  $m$ 
```

- (a) How many comparisons does it make on an array of size n ? [1]
- (b) How many times does m change on line 3 if A is sorted in increasing order? And if it is sorted in decreasing order? [2]

(c) What is its worst-case running time?

[1]

3

6 marks

Suppose that executing line k once takes c_k units of time.

```
1  $s \leftarrow 0$ 
2 for  $i = 1, \dots, n$  do
3   for  $j = 1, \dots, n$  do
4      $s \leftarrow s + A[i] \cdot A[j]$ 
5 return  $s$ 
```

(a) How many times is each line executed? (Count one execution of a **for** line per iteration.) [3]

(b) Write the total time $f(n)$ in terms of $c_1, \dots, c_5$ and n . [2]

(c) Explain why the running time is $\Theta(n^2)$ whatever the values of the c_k . [1]

4

6 marks

Algorithm FIND looks for a value α in A .

```
1 for  $i = 1, \dots, n$  do
2   if  $A[i] = \alpha$  then return  $i$ 
3 return NO
```

(a) Give an input of size n on which it makes exactly one comparison. [1]

(b) How many comparisons does it make if α first appears at position k ? [1]

(c) How many comparisons does it make if α is not in A ? [1]

(d) What is its worst-case running time? [1]

(e) Explain why we call the running time $\Theta(n)$, even though some inputs take only $\Theta(1)$ time. [2]

5

6 marks

```
1 for  $i = 1, \dots, n$  do
2   for  $j = i + 1, \dots, n$  do
3     for  $k = j + 1, \dots, n$  do
4       constant-time step
```

(a) How many times does line 4 run when $n = 4$? [1]

(b) Find the number of times line 4 runs, as a formula in n . [3]

(c) Give its Θ class. [1]

(d) Show that the count is at least $\frac{1}{24}n^3$ for $n \geq 4$. [1]

6

6 marks

Algorithm ALLDISTINCT checks whether the entries of A are all different.

```

1  for  $i = 1, \dots, n$  do
2      for  $j = i + 1, \dots, n$  do
3          if  $A[i] = A[j]$  then return NO
4  return YES

```

- (a) Describe the inputs that make the most comparisons, and give that number. [2]
- (b) Give an input of size $n \geq 2$ that makes only one comparison. [1]
- (c) What is its worst-case running time? [1]
- (d) Give an input of size n on which it makes exactly $n - 1$ comparisons. [2]

7

6 marks

```

1   $i \leftarrow 1$ 
2  while  $i \leq n$  do
3      constant-time step;  $i \leftarrow 2i$ 

```

- (a) How many times does the loop body run when $n = 1$, $n = 8$ and $n = 10$? [2]
- (b) Give the number of iterations as a formula in n . [2]
- (c) What is the running time in Θ notation? [1]
- (d) What changes if line 3 uses $i \leftarrow 3i$ instead? [1]

8

6 marks

```

1  for  $i = 1, \dots, n$  do
2       $j \leftarrow i$ 
3      while  $j \leq n$  do
4          constant-time step;  $j \leftarrow j + i$ 

```

- (a) How many times does the **while** loop body run for $i = 1$, for $i = 2$, and for $i = n$? [3]
- (b) Write the total number of steps as a sum over i . [1]
- (c) Give the running time in Θ notation, with a reason. [2]

9

6 marks

We want the prefix sums $B[k] = A[1] + A[2] + \dots + A[k]$ for every $k = 1, \dots, n$.

- (a) An algorithm computes each $B[k]$ separately by adding up $A[1], \dots, A[k]$ from scratch. How many additions does it make in total, and what is its running time? [2]
- (b) Write pseudocode for an algorithm that computes every $B[k]$ in $\Theta(n)$ time. [2]

- (c) Explain why your algorithm runs in $\Theta(n)$ time. [1]
- (d) For $n = 1000$, roughly how many additions does each algorithm make? [1]

10

7 marks

A is sorted in increasing order. Algorithm SORTEDZEROPAIR uses two indices.

```

1   $\ell \leftarrow 1; r \leftarrow n$ 
2  while  $\ell < r$  do
3       $s \leftarrow A[\ell] + A[r]$ 
4      if  $s = 0$  then return YES
5      else if  $s < 0$  then  $\ell \leftarrow \ell + 1$  else  $r \leftarrow r - 1$ 
6  return NO

```

- (a) Trace it on $A = [-7, -3, -1, 2, 3, 6]$. List the pairs (ℓ, r) it tries and its output. [2]
- (b) Explain why the loop runs at most $n - 1$ times, and give the running time. [2]
- (c) For $n = 1000$, compare the worst-case number of sums it computes with ZEROPAIR from question 1. [2]
- (d) Why does the algorithm need A to be sorted? [1]

11

8 marks

Exam style.

```

1   $c \leftarrow 0$ 
2  for  $i = 1, \dots, n$  do
3      for  $j = 1, \dots, i$  do
4          if  $A[j] > A[i]$  then  $c \leftarrow c + 1$ 
5  return  $c$ 

```

- (a) Describe in words what the value returned counts. [2]
- (b) Find the exact number of comparisons on an array of size n . [2]
- (c) What is the worst-case running time? [1]
- (d) Which inputs give the largest possible return value, and what is it? [2]
- (e) Does the running time depend on the input? Explain. [1]

Exam style. An algorithm runs these three blocks one after another, where $n \geq 2$.

```
1  for  $i = 1, \dots, n$  do constant-time step
2  for  $i = 1, \dots, n$  do
3     $j \leftarrow 1$ ; while  $j < n$  do constant-time step;  $j \leftarrow 2j$ 
4  for  $i = 1, \dots, \lfloor \sqrt{n} \rfloor$  do
5    for  $j = 1, \dots, \lfloor \sqrt{n} \rfloor$  do constant-time step
```

- (a) Count the steps in each of the three blocks. [3]
- (b) Give the total running time in Θ notation, and say which block decides it. [2]
- (c) How does the answer change if line 3 uses $j \leftarrow j + 2$ instead of $j \leftarrow 2j$? [2]
- (d) Explain why the exact times of the individual steps do not affect your answer to (b). [1]

End of questions. Worked answers and the mark scheme are in the matching answers PDF.