

Topic 12 · Sheet A: Worked answers and mark scheme

Greatest common divisors, the euclidean algorithm and Bézout's lemma

Suggested time

About 75 minutes

Total marks

74

Questions

12

Builds on

Topics 1 to 11

How marks are awarded. Each mark has a code, as on most UK exam papers.

B A mark for a correct statement or value on its own. It does not depend on any other mark.

M A method mark, for a correct approach, even if the final answer is wrong.

A An accuracy mark, for the correct answer *after* a correct method. It cannot be awarded without the M mark before it.

E An explanation mark, for a clear reason in words. A correct reason that is worded differently scores in full.

For true or false questions, the mark is for the correct decision; where a reason is asked for, it must be the right reason.

1 Divisor sets

4 marks

(a) $\{1, 2, 4, 7, 14, 28\}$. (b) $\{1, 3, 5, 9, 15, 45\}$. (c) $\{1\}$, so **yes**, they are relatively prime. (d) $\gcd(0, 17) = 17$ (every number divides 0) and $\gcd(0, 0) = 0$ by definition.

B1 each part correct

4

Why this question Divisors come in pairs d and n/d , which makes listing $D(n)$ quick: check up to $\sqrt{n}$ and pair off.

2 Greatest common divisors by factoring

4 marks

(a) $360 = 2^3 \cdot 3^2 \cdot 5$ and $84 = 2^2 \cdot 3 \cdot 7$, so $\gcd = 2^2 \cdot 3 = 12$. (b) $2^3 \cdot 3^2 = 72$. (c) 17, since $51 = 3 \times 17$.

M1	(a) both factorisations	1
A1	(a) 12	1
B1	(b)	1
B1	(c)	1

Why this question Take the smaller power of each prime. This is fast for small numbers, but factoring large numbers is hard, which is why the euclidean algorithm matters.

3 The slow algorithm

6 marks

n	A_n	B_n	action
0	30	12	subtract
1	18	12	subtract
2	6	12	swap
3	12	6	subtract
4	6	6	subtract
5	0	6	swap
6	6	0	halt, $N = 6$

(b) **4 subtractions and 2 swaps.** (c) $\gcd(30, 12) = 6$.

M1	(a) subtracting when $A_n \geq B_n$ and swapping otherwise	1
A3	(a) all rows correct. A2 for one error; A1 for two	3
B1	(b)	1
B1	(c)	1

Common mistake Row 4: when $A_n = B_n$ the rule is still “subtract”, since the test is $A_n \geq B_n$.

4 The fast algorithm

6 marks

n	A_n	B_n	q_n	r_n
0	252	198	1	54
1	198	54	3	36
2	54	36	1	18
3	36	18	2	0
4	18	0	—	—

So $\gcd(252, 198) = 18$. (b) $N = 4$. (c) $1 + 3 + 1 + 2 = 7$.

M1	(a) pivoting forward: $A_{n+1} = B_n, B_{n+1} = r_n$	1
A2	(a) table. <i>A1 for one error</i>	2
A1	(a) gcd 18	1
B1	(b)	1
B1	(c)	1

Why this question Part (c) uses a key fact about the fast algorithm: each q_n counts the subtractions that one division replaces.

5 Division with remainder

6 marks

(a) $q = 14, r = 2$. (b) $q = 22, r = 2$, since $22 \times 56 = 1232$. (c) $q = 0, r = 56$. (d) $q = 27, r = 0$. (e) $a = 8 \times 13 + 5 = 109$. (f) $\gcd(1234, 56) = \gcd(56, 2) = 2$.

B1	each part correct	6
-----------	-------------------	---

Common mistake In (c), giving $r = 1234$ or a negative remainder. When $a < b$, the only choice is $q = 0$ and $r = a$.

6 Rules for gcds

6 marks

(a) **91, 91 and 1**. (b) $\gcd(144, 145) = \gcd(144, 1) = 1$, subtracting 144 from 145. (c) **2**. (d) **(35, 64) and (27, 64)**. The others share a factor: 13 divides 91 and 65, and 17 divides 85.

B1	(a) all three	1
B1	(b)	1
B1	(c)	1
B1	(d) each pair correctly classified, with a shared factor where not coprime	3

Why this question Part (b) generalises: $\gcd(n, n+1) = \gcd(n, 1) = 1$, so neighbouring numbers are always relatively prime.

7 Bézout for a small pair

6 marks

(a) $30 = 2 \cdot 12 + 6, 12 = 2 \cdot 6 + 0$. So $q_0 = 2, q_1 = 2, N = 2$.

(b) Starting from $C_2 = 0, D_2 = (-1)^2 = 1$, and using $C_n = q_n C_{n+1} + D_{n+1}, D_n = C_{n+1}$:

n	C_n	D_n
2	0	1
1	1	0
0	2	1

(c) $30 \times 1 - 12 \times 2 = 6 = \gcd(30, 12)$. **It checks.**

B1	(a)	1
M1	(b) correct starting values	1
A2	(b) table. <i>A1 for one error</i>	2
M1	(c) computing $aD_0 - bC_0$	1
A1	(c) equals 6	1

Why this question The backwards algorithm reuses the quotients from the forward run, from the last one back to the first. Keeping both tables side by side avoids mixing up q_n .

8 Bézout for 252 and 198

6 marks

(a) With $q = (1, 3, 1, 2)$, $N = 4$, $C_4 = 0$, $D_4 = 1$:

n	C_n	D_n
4	0	1
3	1	0
2	1	1
1	4	1
0	5	4

For example $C_1 = q_1 C_2 + D_2 = 3 \cdot 1 + 1 = 4$.

(b) $252 \times 4 - 198 \times 5 = 1008 - 990 = 18$. **It checks.**

(c) $x = 4$, $y = -5$.

M1	(a) $C_N = 0$, $D_N = (-1)^N = 1$	1
A3	(a) table. <i>A2 for one error; A1 for two</i>	3
B1	(b)	1
B1	(c)	1

Common mistake Taking $D_N = 1$ every time. The starting value is $(-1)^N$, which is -1 when N is odd (as in question 10).

9 How fast is the fast algorithm?

6 marks

(a) $13 = 1 \cdot 8 + 5$, $8 = 1 \cdot 5 + 3$, $5 = 1 \cdot 3 + 2$, $3 = 1 \cdot 2 + 1$, $2 = 2 \cdot 1 + 0$. So $q = (1, 1, 1, 1, 2)$ and $N = 5$.

(b) The remainders run down the Fibonacci numbers 55, 34, 21, ...: **10 steps.**

(c) **1 step**: $144 = 12 \times 12 + 0$.

(d) $2 \log_2(144) + 1 \approx 15.3$, so **at most 15 steps.**

M1	(a) the divisions	1
A1	(a) the q_n	1
A1	(a) $N = 5$	1
B1	(b)	1
B1	(c)	1
B1	(d)	1

Why this question Consecutive Fibonacci numbers are the worst case: every quotient is 1, so no division ever saves more than one subtraction.

10 Inverses modulo n from Bézout

6 marks

(a) Fast: $q = (2, 1, 1, 1, 2)$, $N = 5$, $\gcd 1$. Backwards from $C_5 = 0$, $D_5 = (-1)^5 = -1$:

n	5	4	3	2	1	0
C_n	0	-1	-1	-2	-3	-8
D_n	-1	0	-1	-1	-2	-3

So $c = -8$, $d = -3$. Check: $21(-3) - 8(-8) = -63 + 64 = 1$.

(b) $21 \times (-3) \equiv 1 \pmod{8}$, so $x \equiv -3$: $[5]_8$. Check: $105 = 13 \times 8 + 1$.

(c) $-8 \times (-8) = 64 \equiv 1 \pmod{21}$, so $[8]_{21}$. Check: $64 = 3 \times 21 + 1$.

M1	(a) forward run	1
M1	(a) backwards run with $D_5 = -1$	1
A2	(a) c and d with check. <i>A1 for one error</i>	2
B1	(b)	1
B1	(c)	1

Why this question This is the systematic method for linear congruences. Once you have $ad - bc = 1$, reading it modulo b or modulo a hands you an inverse straight away.

11 Exam style: a larger pair

9 marks

- (a) $1071 = 2 \cdot 462 + 147$, $462 = 3 \cdot 147 + 21$, $147 = 7 \cdot 21 + 0$. So **gcd = 21**, with $N = 3$.
- (b) $1071 = 3^2 \cdot 7 \cdot 17$ and $462 = 2 \cdot 3 \cdot 7 \cdot 11$, so $\text{gcd} = 3 \cdot 7 = 21$. **It agrees.**
- (c) From $C_3 = 0$, $D_3 = (-1)^3 = -1$: $(C_2, D_2) = (-1, 0)$, $(C_1, D_1) = (-3, -1)$, $(C_0, D_0) = (-7, -3)$. So **$c = -7$, $d = -3$** .
- (d) $1071(-3) - 462(-7) = -3213 + 3234 = 21$. **It checks.**

M1	(a) divisions	1
A1	(a) quotients and remainders	1
A1	(a) gcd 21	1
B1	(b) both factorisations	1
B1	(b) agreement	1
M1	(c) backwards run with $D_3 = -1$	1
A1	(c) the table	1
A1	(c) $c = -7$, $d = -3$	1
B1	(d)	1

Why this question Factorising 1071 by hand takes some searching; the algorithm takes three short divisions. For large numbers the gap is enormous.

12 Exam style: slow, fast and Bézout together

9 marks

- (a) $(13, 5) \rightarrow (8, 5) \rightarrow (3, 5) \rightarrow (5, 3) \rightarrow (2, 3) \rightarrow (3, 2) \rightarrow (1, 2) \rightarrow (2, 1) \rightarrow (1, 1) \rightarrow (0, 1) \rightarrow (1, 0)$. **6 subtractions and 4 swaps.**
- (b) $13 = 2 \cdot 5 + 3$, $5 = 1 \cdot 3 + 2$, $3 = 1 \cdot 2 + 1$, $2 = 2 \cdot 1 + 0$: $q = (2, 1, 1, 2)$, sum **6**, and **$N = 4$** . Both match.
- (c) Backwards from $C_4 = 0$, $D_4 = 1$: $(C_3, D_3) = (1, 0)$, $(C_2, D_2) = (1, 1)$, $(C_1, D_1) = (2, 1)$, $(C_0, D_0) = (5, 2)$. So **$c = 5$, $d = 2$** . Check: $26 - 25 = 1$.
- (d) $13 \times 2 - 5 \times 5 = 1$ gives $5 \times (-5) \equiv 1 \pmod{13}$, so **$[8]_{13}$** . Check: $40 = 3 \times 13 + 1$.

M1	(a) correct subtract and swap steps	1
A1	(a) 6 subtractions	1
A1	(a) 4 swaps	1
B1	(b) sum of quotients	1
B1	(b) N	1
M1	(c) backwards algorithm with $D_4 = 1$	1
A1	(c) the table	1
A1	(c) $c = 5$, $d = 2$	1
B1	(d)	1

Why this question Part (b) checks the two facts stated after the fast algorithm's first example: the number of fast steps counts the swaps, and the quotients count the subtractions.

Total: 74 marks. Spotted a mistake? Report it at unimixedtutor.com and we will fix it.