Editorial - Classroom

Subtask 1: $M = 1$ (4 points)

Write $[i]$ on every writable paper held by student i. Leave the other papers empty.

After the exchange, the headmaster collects all the indices written on the papers. The missing indices are exactly the students who raised their hands. The permutation changes where a record is found, but does not change the record itself.

This uses $C = 1$.

Subtask 2: $N = 2$ (6 points)

There are at most two hand-raising events, at different steps. We try to store each event as a pair $[i, r]$, meaning that student i raised their hand at step r.

For the first event, write its pair on the other student's paper. If the second event occurs while its student's paper contains the first pair, write the second pair on the empty paper. Both events are now recorded.

The only remaining case is when the second student holds the empty paper. We cannot write on it, so mark the existing pair instead: replace $[i, r]$ with $[i + 2, r]$. The marker means that the other student raised their hand in the step that has just finished.

At the next call, the current step number tells us the missing time. If another teacher enters at step R, that time is $R - 1$; if the headmaster enters, it is $M - 1$. Both students have now raised their hands, so a later teacher can write the two ordinary pairs on the two papers. With no marker, the existing pairs already describe all events.

This uses $C = 2$.

Subtask 3: $P[i] = i$ (9 points)

The papers never move. Treat an empty paper as the number zero. At step r, add $r + 1$ to the number on every writable paper.

Let $S = M(M + 1)/2$. A student who never raises their hand ends with S. A student who raises their hand at step r misses exactly the addition of $r + 1$, so ends with $S - (r + 1)$.

Thus, from the final value x, return -1 if $x = S$, and otherwise return $S - x - 1$. Since $S \le \frac{64 \cdot 63}{2} = 2016$, we can store x using the two integers $\lfloor x/64 \rfloor$ and $x \bmod 64$.

This uses $C = 2$.

Subtaks 4: At most one student raising hand at each step (25 points)

Solution 4a ($C \leq 11$, 8 points)

Reserve one bit per step on every paper, initially zero. At step r, if student x raises their hand, set this step's bit on exactly x writable papers. If nobody raises their hand, set it on all N papers.

There are enough writable papers: when someone raises their hand, there are $N-1$ of them, and $0 \leq x \leq N-1$. Never change bits belonging to earlier steps.

The number of set bits for a step is unchanged by exchanges. At the end, a count of N means no event; any other count is the index of the student who raised their hand.

Pack six bits into each integer. Missing entries of a paper represent zero bits; only extend writable papers. This uses $C \leq \lceil M/6 \rceil \leq 11$.

Solution 4b ($C \leq 4$, 13.75 points)

Whenever student i raises their hand at step r, write $[i, r]$ on an empty writable paper. Keep all existing records.

This fails only if the sole remaining empty paper is held by the student raising their hand. Indeed, with at most one event per step, each earlier event has filled one paper. If no empty paper is writable, there must already be $N-1$ recorded events, and the current event is the last possible one.

In this case, append the new pair to any writable paper. No more events can occur, so no paper ever needs more than two pairs. The headmaster reads all pairs, wherever they have moved.

This uses $C \leq 4$.

Solution 4c ($C \leq 3$, 18.75 points)

Use the previous method until its exceptional case. At that point, the records already contain $N-1$ distinct student indices. The remaining student's index can be recovered as the missing index.

Therefore, append only the current step r to a writable paper. The unique paper of length three contains an ordinary pair followed by this extra time. The headmaster reads all ordinary pairs and assigns the extra time to the one missing student.

This uses $C \leq 3$.

Solution 4d ($C \leq 2$, 25 points)

Again use one pair per event until the exceptional case. Let i be the student raising their hand now, and let $k = (i+1) \bmod N$. Student k already has a record $[k, t]$. Its paper is writable because the only paper we cannot change is the empty one held by i.

Replace $[k, t]$ with $[N, t]$. The value N is a valid integer and is not a student index. It marks the exceptional case without increasing the paper length.

At the next call, two indices are missing from the ordinary records: i and k. For $N > 2$, their relation

$k = (i + 1) \bmod N$ identifies which is which. Student k raised their hand at the stored time t, while student i raised their hand at the previous step. A later teacher can restore both pairs because every student has already raised their hand. The headmaster can instead decode them directly.

For $N = 2$, the two missing indices do not identify their order. Use the marker from Subtask 2, which keeps the first student's index. This distinction is necessary.

This uses $C \leq 2$.

Subtask 5: No additional constraints

Solution 5a ($C = 63$, 4 points)

Maintain an array of N answers, using zero for an unknown event and $r + 1$ for an event at step r.

At each step, take the maximum at each index over all papers, treating empty papers as all zeroes. Add the current events to this array, then copy it to every writable paper. At least one such paper exists, so a complete copy of the history survives. Older copies never introduce a false event, and taking maxima recovers every recorded event.

The headmaster takes the same maxima and subtracts one from each entry. This uses $C = N \leq 63$ and $O(N^2)$ time per call.

Another method uses one entry per step. At step r, write $i + 1$ in entry r of each writable paper held by student i, leaving that entry zero on the other papers. Missing entries count as zero. The headmaster finds the missing student indices separately for every step, as in Subtask 1. This uses $C = M \leq 63$.

Solution 5b ($C \leq 13$, 26 points)

For each of the first $M - 3$ steps, reserve one bit on each paper. Set the bit if the paper is writable, and otherwise leave it zero. Unlike the bit-count solution, set it on every writable paper: once we know P, we can trace each bit back to its student.

For each of the last three steps, reserve a full integer instead. At step r, write $i + 1$ in that step's entry on every writable paper held by student i. Zero means that the entry was not written. If $M \leq 3$, use only these full entries.

The last three entries reveal the events in those steps by their missing indices. They also reveal P. Two nonzero entries for consecutive steps on the same paper give a pair $i, P[i]$. A nonzero entry for the final step and the paper's final holder give another such pair.

For any fixed i, there are three chances to observe the edge from i to $P[i]$. A hand raise by i can spoil at most one chance, and a hand raise by $P[i]$ can spoil at most one more. Both students raise their hands at most once, so at least one chance remains. Thus every value of P is recovered.

A paper held by student i at step r ends at student $P^{M-r}[i]$, where the exponent means applying the permutation that many times. We can now read the bit for every earlier student and step. A zero bit identifies a hand raise.

The length is at most $\lceil (M - 3)/6 \rceil + 3 \leq 13$ when $M > 3$, and at most M otherwise. Decoding takes

$O(NM)$ time by tracing the papers backwards one step at a time.

Solution 5c ($C = 6$ or $C = 5$, 47 points or 50 points)

We would like to keep event records $[i, r]$, but many students may raise their hands together. If only a few papers are writable, there may be nowhere to put all these records.

At a step with at most $N/2$ hand raises, record the students who raise their hands. At a step with more than $N/2$ hand raises, record the students who do not. Call the latter records **complement records**. There can be at most one such step, since students never raise their hands twice.

Give each paper two record slots. The first slot holds only an ordinary event record. The second slot can hold either type, so it also stores a type flag. An occupied first slot needs two integers, and an occupied second slot needs three. Unused slots can be represented with a zero student field, using $i + 1$ for actual indices.

Place ordinary records into writable first slots whenever possible, then into writable second slots. Place complement records only into second slots.

To see that there is enough space, let t be the number of hand raises now and h the number of earlier records. We always have $h + t \leq N$: each earlier step stored no more records than its number of events.

If $t \leq N/2$, the $N - t$ writable papers have $2(N - t)$ slots. At most h are occupied, so at least $2(N - t) - h \geq t$ are available. Ordinary records can use either slot.

If $t > N/2$, let $k = N - t$. There were fewer than $N/2$ events in total before this step. All earlier records therefore fit in first slots: before each such insertion, an empty writable first slot existed. Consequently, every second slot is still empty. Put one complement record on each of the k writable papers.

The headmaster reads ordinary records directly. If complement records exist, their common time identifies the large step, and all indices absent from their list raised their hands at that step.

Using a separate type flag for both slots gives the simpler $C \leq 6$ version. Restricting the first slot as above gives $C \leq 5$.

Solution 5d ($C \leq 3$, 75 points)

Instead of saving every event immediately, detect it at the next call. Use two kinds of records:

- A temporary record (i, r) says that student i did not raise their hand at step r.
- A final record (i, r) says that student i raised their hand at step r.

Store the student, the step, and a flag distinguishing these two kinds, using three integers.

At step zero, write a temporary record $(i, 0)$ on every writable paper held by student i.

At each later step R, first read all papers. For every student i whose paper is writable, choose its new contents using only this input snapshot:

- If a final record for i exists, copy it to student i's paper.
- Otherwise, if a temporary record $(i, R - 1)$ exists, write the temporary record (i, R).
- Otherwise, write the final record $(i, R - 1)$.

The third case means that i could not write a record in the previous step. Therefore i raised their hand then. Since each student raises their hand only once, their paper is writable now and at every later step. Their final record can always be copied back to them before the next exchange.

Old temporary records may survive on papers that cannot be changed, but their exact step numbers prevent them from being mistaken for records from the previous step.

The headmaster runs the same update as an extra step $R = M$, with every paper treated as writable. The final records give the answer; all remaining temporary records mean that the student never raised their hand. This last update is only a local computation, so its records need not be encoded onto papers.

Solution 5e ($C = 2$, 100 points)

Keep the previous solution, but replace the exact time in a temporary record with its remainder modulo three. Write $U(i, r \bmod 3)$ for a temporary record, and $F(i, r)$ for a final record. We will also use a special record $D(i)$ to repair one exceptional case.

The normal update is unchanged: a final record for i takes priority; otherwise, a temporary record with the previous step's remainder means that i is still waiting to raise their hand. If neither exists, finalise i at the previous step.

The difficulty is that an old temporary record can survive long enough for its remainder to match again. We must detect this before it causes a wrong answer.

Suppose that, at step R, student i raises their hand, and a paper held by a student raising their hand contains $U(i, R \bmod 3)$. This may be student i's own paper. This record will survive the current step. At step $R + 1$, it would incorrectly suggest that i did not raise their hand at step R.

These are exactly the cases we need to repair. A paper that survives unchanged for several exchanges must be held by a student raising their hand at every intervening step. If the stored remainder can be mistaken for the current one, at least three steps have passed. This gives us two students who have already raised their hands and whose papers are now writable.

We can find them without knowing all of P. Student i had not raised their hand before step R, so they wrote a fresh temporary record at step $R - 1$. It is now at $P[i]$, whose student has already raised their hand and is writable. The stale copy mentioning i is on a paper that is not writable. Thus, among writable holders, the record mentioning i identifies $P[i]$.

The same argument continues along the path. Each intermediate student who raised their hand before step $R - 1$ wrote their final record in the previous step; its writable holder identifies the next student. The student who raised their hand at step $R - 1$ could not write a record then, so the path stops there. To implement this, build a link from each stored index to its current holder, considering only writable holders. These links need not recover P everywhere, but they give the correct path starting from each problematic i. Follow that path until no link exists. Let a and b be the last two students reached after i. Student a raised their hand at step $R - 2$, and student $b = P[a]$ raised their hand at step $R - 1$. Both papers are writable now. The path has at least these two students because we use remainders modulo three.

After making the normal updates, replace the paper at a with $D(i)$ and the paper at b with $D(a)$. We temporarily borrow the space for their final records. The markers also preserve the information needed to recover the event at i.

At the next call, let the current step be $S = R + 1$. Read the special records from the input snapshot:

- The marker $D(i)$ has moved from a to b. There is also an ordinary temporary record mentioning i: the old record that caused the problem. Restore $F(i, S-1)$ on student i's paper and $F(b, S-2)$ on the marker holder's paper.
- The marker $D(a)$ has moved from b to $c = P[b]$. No ordinary record mentions a. Restore $F(a, S-3)$ on student a's paper and $F(c, S-1)$ on the marker holder's paper. Student c held the old record and raised their hand at step R.

Thus a marker $D(x)$ is decoded by checking whether any non-special input record mentions x. If yes, use the first rule; otherwise, use the second. This also handles $c = i$, when the path closes back at the problematic student.

Every paper changed by these repair rules belongs to a student who has already raised their hand, so it is writable. Separate problematic records follow disjoint paths: on each path, the intermediate students have already raised their hands, and so cannot start another problem in the same step. Repairs from the previous step also concern students who have already raised their hands. We can therefore handle all cases in one update.

In implementation, use the same order as the reference solution: compute normal updates from the input snapshot, create markers for new problems, then apply repairs for markers in the input. Never use a newly written record as input to another decision in the same call. Initialise step zero separately, and let the headmaster run the update at step M with no hands raised.

It remains to encode the records. The final step $M - 1$ is detected only by the headmaster, so teachers only need to store final times 0 through 61. The following encoding fits every paper into at most two integers:

- Empty record: $[\,]$.
- $F(i, t)$: $[i, t]$, where $0 \le t \le 61$.
- $U(i, 0)$: $[i, 62]$.
- $U(i, 1)$: $[i, 63]$.
- $U(i, 2)$: $[63, i]$.
- $D(i)$: $[i]$.

Student indices are at most 62, so these representations are distinct, and every integer lies in $[0, 63]$. The headmaster's final update operates on decoded records and does not need to encode its result.

A direct implementation scans the papers for each student and takes $O(N^2)$ time per call, with $O(N)$ additional memory. This is sufficient for $N, M \le 63$. Every returned paper has length at most two, giving full points.