Prob. 1	Prob. 2	Prob. 3

Problem 1.

We can first define what the optimal strategy knowing T is :

- if $T < N$, then we rent skis T times,
- if $N \leqslant T \leqslant 2N$, then we buy cheap skis,
- if $2N < T < 3N$, we buy cheap skis and then rent skis for $T - 2N$ times,
- if $T \geqslant 3N$, we buy the expensive skis which will last forever.

We are going to prove that we can have a competitve ratio slightly better than two ($2(1 - 1/2N)$) and that there is no online strategy with a better competitive ratio. First, let's describe our online strategy :

- we begin by renting skis N-1 times,
- then we buy cheap skis,
- and finally if $T > 3N - 1$ we buy the expensive ones.

We denote respectively by $OPT(T)$ and $ONL(T)$ the price paid in the optimal strategy and our online strategy.

If $T \leqslant N - 1$ then we have $\frac{ONL(T)}{OPT(T)} = 1$
If $N \leqslant T \leqslant 3N - 1$ then we have $\frac{ONL(T)}{OPT(T)} \leqslant \frac{2N-1}{N} < 2(1 - 1/2N)$
If $T \geqslant 3N$ then we have $\frac{ONL(T)}{OPT(T)} \leqslant \frac{4N-1}{2N} = 2(1 - 1/2N)$
Therefore, our strategy competitve ratio is $2(1 - 1/2N)$.

We now need to prove that all other possible strategies have higher competitive ratio; actually we are going to prove that all these strategies have a competitive ratio of at least 2. First, we can note that buying a pair of skis before time N-1 (let's say at time K) is going to lead to a competitive ratio higher or equal than $\frac{N+K}{K+1}$. This function of K is decreasing so it is higher to $\frac{2N-2}{N-1} = 2$.

On the contrary, if we choose to rent our skis for more than N-1 times, we have to distinguish two cases :

- at some point along the way, we buy the cheap skis. It is obvious that if T becomes very big ($10N$ for instance), we will have to buy the expensive skis too. Then our competitive ratio is a least $\frac{N+N+2N}{2N} = 2$.

- we never buy the cheap skis but directly the expensive ones. Now we need to consider when we are buying these skis. If it happens before T = 2N, when we buy them we have a competitive ratio higher than 2. If it happens at T=2N or after, at time 4N our ratio is going to be higher than 2.

We have just etasblished that the number of rentings needs to be exactly N-1. Then at time N we buy the cheap skis. The last question that needs to be answered is : what happens at T=3N-1 when my cheap skis are just broken? At this point we need to buy the expensive skis because if we don't when we will buy them in the future (and we will have to) our competitive ratio is going to be more than 2.

Problem 2.

Let's call our 2^{nd} access-MTF MTF2 and compute the competitive ratio with the static algorithm. Let ϕ_x be the number of items that precede x in the MTF2 list, but follow it in the optimal static list (the same as in class), b_x the bit associated to x and j_x the position of x in the optimal static list. For the potential function, we want to count the contribution of an item x if his associated bit is 0 and add 1. Our potential thus becomes :

$$\phi = \sum 2 * \phi_x + b_x \times (j_x - \phi_x)$$

.

We need to compute the amortized time for the two cases.

First access : Nothing moves but the bit of x that goes from 0 to 1.

$$\begin{aligned} \textit{real cost} + \Delta\phi &= i + (j - \phi_x) \\ \textit{real cost} + \Delta\phi &\leq 2 \times j \end{aligned}$$

Second access : Variations in potential come from three sources : - ϕ_x is reduced to 0 - b_x goes from 1 to 0 thus removing j_x to the potential - the $(i - \phi_x)$ elements in front of x move each inducing a possible increase in the potential of 2.

$$\begin{aligned} \textit{real cost} + \Delta\phi &\leq i - (j - \phi_x) - 2 \times \phi_x + 2 \times (i - \phi_x) \\ \textit{real cost} + \Delta\phi &\leq 3 \times (i - \phi_x) - j \\ \textit{real cost} + \Delta\phi &\leq 2 \times j \end{aligned}$$

We can deduce that the competitive ratio of MTF2 with the optimal static list is 2.

For MTFk let n_x be the number of times x has been accessed since his last move to the front. With the potential $\phi = \sum k * \phi_x + n_x \times (j_x - \phi_x)$ we can show that MTFk has the same competitive ratio as MTF2.

Problem 3.

1. Let use the strategy where at any step n (n starts at 0) the position of the dog is given by $x_n = (-2)^n$ (except if the dog has found the bone, in this case the dog stops at the bone).

The total distance t_n walked by the dog since the beginning of the search until the step n is given by the relation

$$t_n = \begin{cases} 1 & \text{if } n \text{ is } 0 \\ t_{n-1} + 2^{n-1} + 2^n = t_{n-1} + 3 * 2^{n-1} & \text{if } n > 0, \text{ bone not found at step n} \\ t_{n-1} + 2^{n-1} + |x| & \text{if the bone was found at step } n \end{cases}$$

Thus, we have

$$t_n = \begin{cases} 3 * (2^n - 1) + 1 & \text{for every } n \text{ before the dog finds the bone} \\ 3 * (2^{n-1} - 1) + 2^{n-1} + |x| & \text{at the step where the dog finds the bone} \end{cases}$$

(Even if the bone is found at the step 0).

If $x > 0$ or $x < 0$, the dog finds the bone at the step N such that $2^{N-2} < |x| \leq 2^N$ (the dog didn't find the bone at the second to last step).
Then in this case the total walked distance is:
$T = t_N = 3 * (2^{N-1} - 1) + 1 + 2^{N-1} + |x| < 3 * (2 * |x| - 1) + 1 + 2 * |x| + |x| = 9 * |x| - 2$

Thus, the total walked distance is bounded by $9 * |x| - 2$. Our algorithm is competitive with the best algorithm with a ratio of 9.

2. This time we have $m > 2$ roads crossing at the same point. We index the roads by $0..m-1$. We note by $q(n,m)$ the quotient of n divided by m, and $r(n,m)$ the remainder of the division of n by m.

Our dog will search the bone by making circles, going from road r to road $r + 1[m]$ at each step. We are going to use quite the same strategy as before: we will double the distance each time the dog passes by the road 0. Let $X_n = (r_n, x_n)$ be the tuple containing the index of the road on which the dog is and his absciss on this road.
The dog will change of road between each step, and double the absciss if he is on the road 0: so we have $r_n = r(n,m)$ and $x_n = 2^{q(n,m)}$.

Let t_n be the distance that the dog has walked since the beginning of the search. We have the relation: $t_n = \begin{cases} 1 & \text{if } n \text{ is } 0 \\ t_{n-1} + 2^{q(n-1,m)} + 2^{q(n,m)} & \text{if } n > 0, \text{ bone not found at step n} \\ t_{n-1} + 2^{q(n-1,m)} + x & \text{if the bone was found at step } n \end{cases}$

At the step n, the dog has made $q(n,m)$ times a complete lap.

- To get at the origin (absciss 0 on the road 0) after the $q(n,m)$ laps, the dog has walked the distance:

$\sum_{l=0}^{q(n,m)-1} 2*m*2^l = 2m*(2^{q(n,m)}-1)$
(because there are m roads, and the dog must walk back and forth on the roads).

- Then, to be on the correct road, at the correct absciss, the dog must walk
$\begin{cases} (2*r(n,m)+1)*2^{q(n,m)} & \text{(if he hasn't found the bone at the step } n \\ 2*r(n,m)*2^{q(n,m)}+x & \text{if he has found the bone at the step } n \end{cases}$

So we have:

$$t_n = \begin{cases} 2m*(2^{q(n,m)}-1)+(2*r(n,m)+1)*2^{q(n,m)} & \text{for every } n \text{ before the dog finds the bone} \\ 2m*(2^{q(n,m)}-1)+2*r(n,m)*2^{q(n,m)}+x & \text{at the step where the dog finds the bone} \end{cases}$$

Let suppose that the bone is on the road r, then the dog finds it at a step $N = m*k+r$ such that $2^{k-1} < x \leq 2^k$.
The distance that the dog has walked to find the bone is:
$T = t_N = 2m*(2^k-1)+2r*2^k+x < 4mx-2m+4rx+x = (4(m+r)+1)x-2m$
Thus, we always have $T = t_N < (8m-3)x-2m = O(mx)$.

If we use the fact that $x \leq 2^k$, we deduce that:
$T = t_N = 2m*(2^k-1)+2r*2^k+x \geq 2m(x-1)+2rx+x = (2(m+r)+1)x-2m \geq 2m(x-1)$.
So, $T = \Omega(mx)$

As a conclusion, we have a competitive ratio of $\Theta(mx)$.