Problem 5.01

The following claim can be disproved using a counter-example. Let $n = 6$ and $k = 2$. Since n is even, and $2 \leq k \leq n-2$, the preconditions have been satisfied. If we try and compute it we get:

$$\begin{aligned}\binom{6}{2} &= \frac{6!}{4! \cdot 2!} \\ &= \frac{6 \cdot 5 \cdot 4 \cdot 3 \cdot 2}{4 \cdot 3 \cdot 2 \cdot 2} \\ &= \frac{6 \cdot 5}{2} \\ &= 3 \cdot 5 \\ &= 15\end{aligned}$$

Thus we have disproved the statement since $\binom{n}{k}$ is not even when $n = 6$ and $k = 2$

Problem 5.02

The claim can be proven using a direct proof. We need to show that $n^2 = 2\binom{n}{2} + n$. We have:

$$\begin{aligned}2\binom{n}{2} + n &= 2 \cdot \frac{n!}{(n-2)!2!} + n \\ &= 2 \cdot \frac{n(n-1)(n-2)!}{(n-2)!2} + n \\ &= n(n-1) + n \\ &= n^2 - n + n \\ &= n^2\end{aligned}$$

Note that this is only possible if $n \geq 2$ since if $n < 2$ then the cancellations made in this proof would not be possible, since we have to deal with negative factorials.

Problem 5.03Y

The following claim can be disproved using a counter-example. Let us use the following parameters

```
studnts = [0]    begin_s = 0    end_s = 0    target = 0
```

These conditions satisfies the preconditions as $0 \leq begin\_s \leq end\_s$ and the $target$ is within the range. Since $low := begin\_s$ and $high := end\_s$ and by line 5 of the code we can see that the condition of the while loop states that $low < high$. Now, since $0 < 0$ is not true, the program never enters the while loop and returns -1. Thus the statement is not true.

Problem 5.03N

Assume some arbitrary array of $studnts$ and some range $begin\_s > 0$ and $end\_s < studnts.length$ such that the $target$ is not within the range of $[begin\_s - end\_s]$. This leaves us with two different cases. Recall that by line 6 of the code that $mid = \dfrac{high + low}{2}$

Case 1: target $<$ students[mid] By line 8 of the code that $high = mid$. Thus, as the program iterates through the while loop, that the value of $high$ will slowly start converging towards the value of low, then we can say that the $[low - high]$ range will never contain $target$.

Case 2: target $>$ studnts[mid] by line 10 of the code that $low = mid$. Thus as the program iterates through the while loop, that the value of low will slowly start converging towards the value of $high$ then we can say that the $[low - high]$ range will never contain $target$.
We can also use the proposition 5.03P, it states that for any $n, k \in \mathbb{Z}^+$ that $n < 2^k$ is true. We can rewrite this statement to look like $\dfrac{n}{2^k} < 1$. We can say that n is the size of the range $[begin\_s - end\_s]$, and that k is the number of iterations we go throughout the function. From here we can see that each iteration, since either $high$ or low, gets replaced with the value of mid that the range eventually converges into less than 1 element. So given the two cases and this fact, we can say that if the $target$ is not within the range, that when the function does return, it will return -1

Problem 5.03T

We can use the array $[0, 1]$. We can have $low = 0$ and $high = 1$ as our starting values, and our $target = 1$. We have it such that $\dfrac{low + high}{2} = \dfrac{0+1}{2} = \dfrac{1}{2}$. Since, we are flooring the function, we have it such that $mid = 0$ and that $studnts[mid] = 0$. Since $0 < 1$ based on line 10 of the code, we replace low with mid. So since, low stays the same per iteration, and it is always true that $low < high$, we stay in the while loop forever, and the function is never terminated.