In [7]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [8]:
```
sequence1 = "AAAAGGGTGTCTGTAGTGAGGGATGTGCCCATGAGTAGCTAGTCAGTCACTGCTACG
sequence2 = "AGAGATGTCGTGHTGTYHTABGTAGGAGCTCGYTGRGHGTGAFHGTGAGTGGAGAGT
```

In [9]:
```
def clean_sequence(sequence):
    """Filters out any non-ACGT characters from a DNA sequence."""
    valid_nucleotides = {'A', 'C', 'G', 'T'}
    cleaned_seq = "".join([n for n in sequence.upper() if n in valid_n
    return cleaned_seq

cleaned_seq1 = clean_sequence(sequence1)
cleaned_seq2 = clean_sequence(sequence2)

print(f"Cleaned Sequence 1 Length: {len(cleaned_seq1)}")
print(f"Cleaned Sequence 2 Length: {len(cleaned_seq2)}")
```

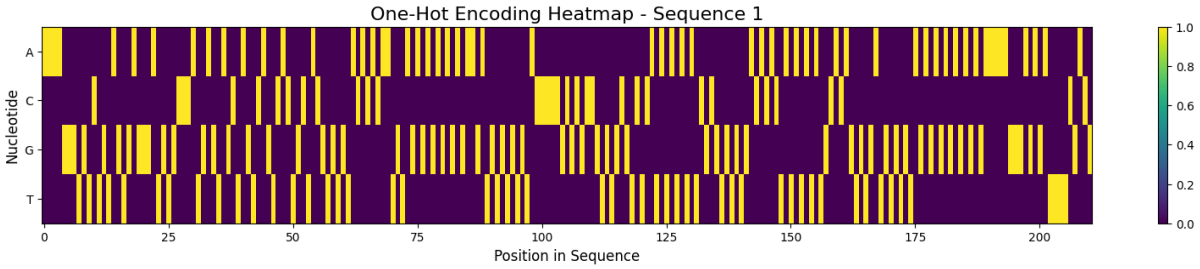
```
Cleaned Sequence 1 Length: 211
Cleaned Sequence 2 Length: 168
```

In [10]:
```
def one_hot_encode(sequence):
    """Converts a clean DNA sequence to a one-hot encoded numpy array.
    encoding_map = {'A': [1, 0, 0, 0],
                    'C': [0, 1, 0, 0],
                    'G': [0, 0, 1, 0],
                    'T': [0, 0, 0, 1]}

    one_hot = [encoding_map[nucleotide] for nucleotide in sequence]
    return np.array(one_hot)

encoded_seq1 = one_hot_encode(cleaned_seq1)
encoded_seq2 = one_hot_encode(cleaned_seq2)

print(f"Encoded Sequence 1 Shape: {encoded_seq1.shape}")
print(f"Encoded Sequence 2 Shape: {encoded_seq2.shape}")
```

```
Encoded Sequence 1 Shape: (211, 4)
Encoded Sequence 2 Shape: (168, 4)
```

In [11]:
```
def plot_one_hot_heatmap(encoded_matrix, title):
    plt.figure(figsize=(20, 3))

    plt.imshow(encoded_matrix.T, cmap='viridis', aspect='auto')


    plt.title(title, fontsize=16)
    plt.xlabel("Position in Sequence", fontsize=12)
    plt.ylabel("Nucleotide", fontsize=12)
```

```
    plt.yticks(ticks=[0, 1, 2, 3], labels=['A', 'C', 'G', 'T'])


    plt.colorbar()

    plt.show()

plot_one_hot_heatmap(encoded_seq1, "One-Hot Encoding Heatmap - Sequenc
```



In [12]:
```
plot_one_hot_heatmap(encoded_seq2, "One-Hot Encoding Heatmap - Sequenc
```