SWETHA NALLAMANGAI K N

21ADR054

Swethanallamangaikn.21aid@kongu.edu

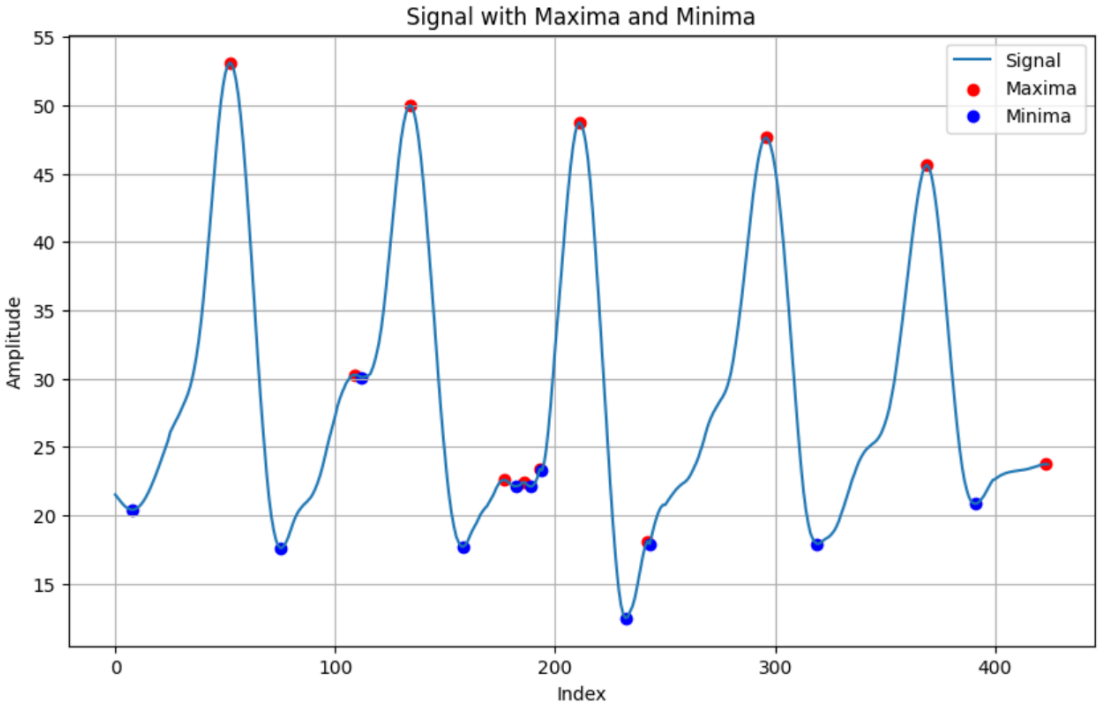
QUESTION-1

PYTHON

DATA-1:

```
import matplotlib.pyplot as plt
def find_peaks(data):
    maxima = []
    minima = []
    n = len(data)
    for i in range(1, n - 1):
        if data[i - 1] < data[i] > data[i + 1]:
            maxima.append(i)
        if data[i - 1] > data[i] < data[i + 1]:
            minima.append(i)
    return maxima, minima
with open('/content/Data_1.txt', 'r') as file:
    data = [float(line.strip()) for line in file.readlines()]
maxima, minima = find_peaks(data)
plt.figure(figsize=(10, 6))
plt.plot(data, label='Signal')
plt.scatter(maxima, [data[i] for i in maxima], color='red', label='Maxima')
plt.scatter(minima, [data[i] for i in minima], color='blue', label='Minima')
plt.title('Signal with Maxima and Minima')
plt.xlabel('Index')
plt.ylabel('Amplitude')
plt.legend()
plt.grid(True)
plt.show()
```

OUTPUT:

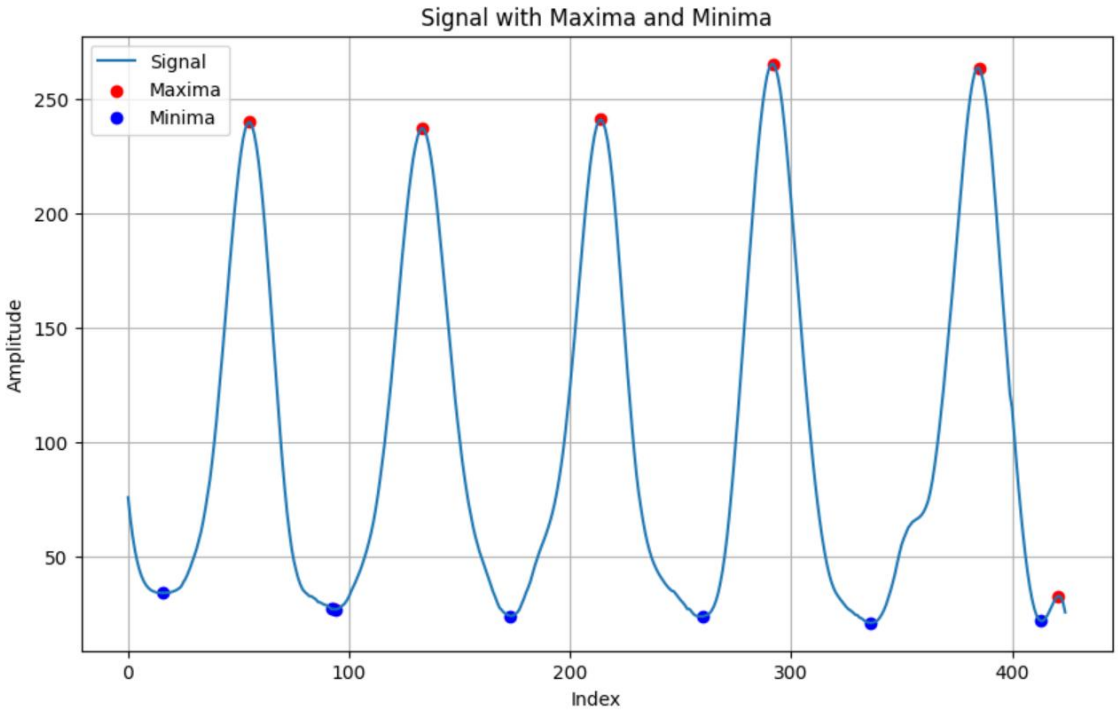


DATA-2 :

```
import matplotlib.pyplot as plt
def find_peaks(data):
    maxima = []
    minima = []
    n = len(data)
    for i in range(1, n - 1):
        if data[i - 1] < data[i] > data[i + 1]:
            maxima.append(i)
        if data[i - 1] > data[i] < data[i + 1]:
            minima.append(i)
    return maxima, minima
with open('/content/Data_2.txt', 'r') as file:
    data = [float(line.strip()) for line in file.readlines()]
maxima, minima = find_peaks(data)
plt.figure(figsize=(10, 6))
plt.plot(data, label='Signal')
```

```
plt.scatter(maxima, [data[i] for i in maxima], color='red', label='Maxima')
plt.scatter(minima, [data[i] for i in minima], color='blue', label='Minima')
plt.title('Signal with Maxima and Minima')
plt.xlabel('Index')
plt.ylabel('Amplitude')
plt.legend()
plt.grid(True)
plt.show()
```

OUTPUT:



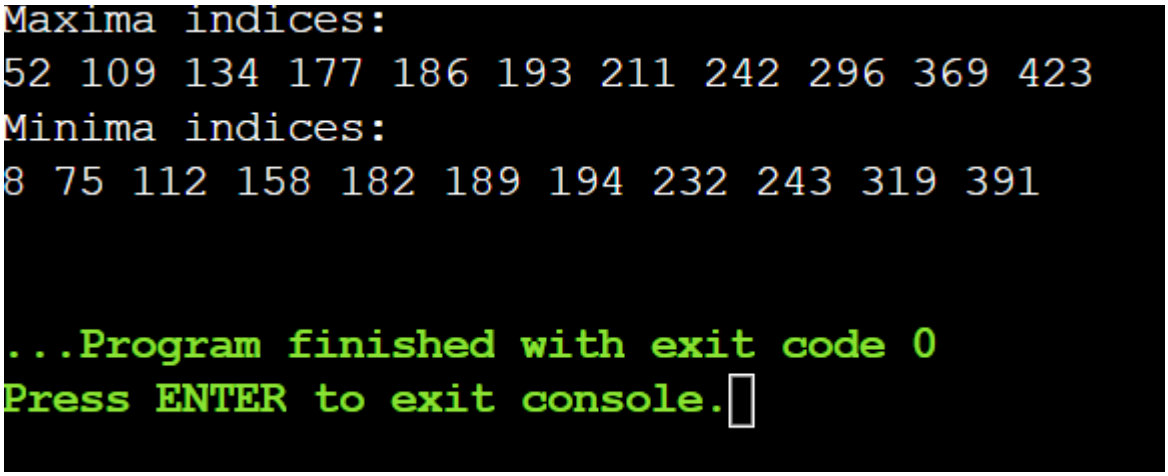
C CODE:

DATA 1:

```
#include <stdio.h>
void find_peaks(double data[], int n, int maxima[], int *max_count, int minima[], int
*min_count) {
    *max_count = 0;
    *min_count = 0;
    for (int i = 1; i < n - 1; i++) {
        if (data[i - 1] < data[i] && data[i] > data[i + 1]) {
            maxima[(*max_count)++] = i;
        }
        if (data[i - 1] > data[i] && data[i] < data[i + 1]) {
            minima[(*min_count)++] = i;
        }
    }
}
int main() {
    FILE *file = fopen("Data_1.txt", "r");
    if (file == NULL) {
        printf("Failed to open file\n");
        return 1;
    }
    double data[1000];
    int n = 0;
    while (fscanf(file, "%lf", &data[n]) != EOF) {
        n++;
    }
    fclose(file);
    int maxima[1000], minima[1000];
    int max_count, min_count;
    find_peaks(data, n, maxima, &max_count, minima, &min_count);
```

```
    printf("Maxima indices:\n");
    for (int i = 0; i < max_count; i++) {
        printf("%d ", maxima[i]);
    }
    printf("\nMinima indices:\n");
    for (int i = 0; i < min_count; i++) {
        printf("%d ", minima[i]);
    }
    printf("\n");
    return 0;
}
```

OUTPUT:



DATA 2:

```
#include <stdio.h>
void find_peaks(double data[], int n, int maxima[], int *max_count, int minima[], int
*min_count) {
    *max_count = 0;
    *min_count = 0;
    for (int i = 1; i < n - 1; i++) {
        if (data[i - 1] < data[i] && data[i] > data[i + 1]) {
            maxima[(*max_count)++] = i;
        }
        if (data[i - 1] > data[i] && data[i] < data[i + 1]) {
            minima[(*min_count)++] = i;
        }
    }
}
int main() {
    FILE *file = fopen("Data_2.txt", "r");
    if (file == NULL) {
        printf("Failed to open file\n");
        return 1;
    }
    double data[1000];
    int n = 0;
    while (fscanf(file, "%lf", &data[n]) != EOF) {
        n++;
    }
    fclose(file);
    int maxima[1000], minima[1000];
    int max_count, min_count;
    find_peaks(data, n, maxima, &max_count, minima, &min_count);
    printf("Maxima indices:\n");
```

```
    for (int i = 0; i < max_count; i++) {
        printf("%d ", maxima[i]);
    }
    printf("\nMinima indices:\n");
    for (int i = 0; i < min_count; i++) {
        printf("%d ", minima[i]);
    }
    printf("\n");
    return 0;
}
```

OUTPUT: