O Notation Examples

O Notation

```
int getThird(int *arr){
    return arr[3];
}
```

O(1)

And what two things are wrong with the function?

In all these examples "n" is the size of the input
  e.g. length of arr

O Notation

```
int getMin(int *arr, int n){
    smallest = arr[0];
    for (int i=0; i < n; ++i)
     {
          if (arr[i] < smallest) {
                smallest = arr[i];
          }
    }
    return smallest ;
}
```

O(n)

O Notation

```
int * f1(int *arr, int n):
    int *result = new int[n];
    for (int i=0; i < n; ++i){
        result[i] = arr[i] - getMin(arr);
    }
    return result;
}
```

$O(n^2)$

O Notation

```
int f2(int *arr, int n){
    if (n > 0){
        return arr[n - 1];
    }
    else{
        return -1;
    }
}
```

O(1)

O Notation

```
double f3(double *arr, int n):
    double diff = 0;
    int ln = n;
    for (int i = 0; i < ln; ++i){
        double total = 0.0;
        for (int j = 0; j < ln; ++j){
            total += arr[j];
        }
        diff += arr[i] - total / ln;
    }
    return diff;
}
```

$O(n^2)$

O Notation

```
double f3(double *arr, int n):
    double diff = 0;
    double avg = 0;
    for (int i=0; i < n; ++i){
         avg += arr[i];
    }
    avg = avg / n;
    for (int j=0; j < n; ++j){
         diff += arr[j] - avg;
    }
    return diff;
}
```

O(n)

O Notation

```
int f4(int *arr, int n, int m){
    if (m > n - 1){
        return 0;
    } else if  (m == n - 1){
        return arr[m];
    }
    else{
        return arr[m] + f4(arr, n, m + 1);
    }
}
```

O(n)

O Notation

```
int f5(int *arr, int n){
    int result = 0;
    int i = 0;
    ln = n;
    while (i < ln / 2){
        result += arr[i];
        i += 1;
        while (i >= ln / 2 && i < ln){
            result += arr[i];
            i += 1;
        }
    }
    return result;
}
```

O(n)

O Notation

```
bool alpha(string s){
  int ln = s.size();
  for (int i = 0; i < ln - 1; ++i){
    if (s[i] > s[i + 1]){
         return false;
    }
  }
  return true;
}
```

best case: ?

average case: ?

worst case: ?

Here n is the length of the String s