Advanced Time Complexity Worksheet

1.
```
for (int i = 1; i < n; i *= 2)
    for (int j = 1; j < n; j += i)
        for (int k = 1; k < j; k *= 2)
            // do something
```
2.
```
for (int i = 1; i < n; i *= 2)
    for (int j = n; j > 0; j /= i)
        // do something
```
3.
```
int count = 0;
for (int i = 1; i <= n; i++) {
    for (int j = i; j <= n; j += i) {
        count++;
    }
}
```
4.
```
for (int i = 1; i < n; i++)
    for (int j = 1; j < i; j *= j)
        // do something
```
5.
```
for (int i = 0; i < n; i++) {
    int j = i;
    while (j > 0) {
        j = j & (j - 1);
    }
}
```
6.
```
void fun(int n) {
    if (n <= 1) return;
    fun(n / 2);
    fun(n / 4);
    fun(n / 8);
}
```
7.
```
void fun(int n) {
    if (n <= 1) return;
    fun(n - sqrt(n));
}
```
8.
```
void fun(int n) {
    if (n <= 1) return;
    for (int i = 1; i < n; i++)
        fun(i);
}
```
9.
```
for (int i = 0; i < n; i++) {
    if (__builtin_popcount(i) % 2 == 0) {
        for (int j = 0; j < i; j++) {
            // do something
        }
```

```
    }
}
10. int x = 1;
while (x < n) {
    x = pow(x, 1.5);
    // do something
}
```