Introduction to Algorithms

Sheet 2

Time Complexity

What is the time complexity of the following algorithms?

Algorithm 1

```
int t = 0;
for (int i = 1; i <= n; i++)
    for (int j = 0; j * j < 4 * n; j++)
        for (int k = 1; k * k <= 9 * n; k++)
            t++;
```

Algorithm 2

```
int x = 0;
for (int i = 4 * n; i >= 1; i--)
  x = x + 2 * i;
```

Algorithm 3

```
int z = 0;
int x = 0;
for (int i = 1; i <= n; i = i * 3) {
  z = z + 5;
  z++;
  x = 2 * x;
}
```

Algorithm 4

```
int y = 0;
for (int j = 1; j * j <= n; j++)
  y++;
```

Algorithm 5

```
int b = 0;
for (int i = n; i > 0; i--)
  for (int j = 0; j < i; j++)
    b = b + 5;
```

Algorithm 6

```
int y = 1;
int j = 0;
for (j = 1; j <= 2n; j = j + 2)
  y = y + i;
int s = 0;
for (i = 1; i <= j; i++)
  s++;
```

Algorithm 7

```
int b = 0;
for (int i = 0; i < n; i++)
  for (int j = 0; j < i * n; j++)
    b = b + 5;
```

Algorithm 8

```
int x = 0;
for (int i = 1; i <= n; i = i * 3) {
  if (i % 2 != 0)
    for (int j = 0; j < i; j++)
      x++;
}
```

Algorithm 9

```
for (int i = 0; i <= n; i++)
    for (int j = 1; j <= i * i; j++)
      if (j % i == 0)
        for (int k = 0; k < j; k++)
          sum++;
```

Algorithm 10

```
int a = 0;
int k = n * n;
while (k > 1) {
  for (int j = 0; j < n * n; j++) {
    a++;
  }
  k = k / 2;
}
```

Algorithm 11

```
int fun(int n) {
  int count = 0;
  for (int i = n; i > 0; i /= 2)
    for (int j = 0; j < i; j++)
      count += 1;
  return count;
}
```

Algorithm 12

```
int fun(int n) {
  int count = 0;
  for (int i = 0; i < n; i++)
    for (int j = i; j > 0; j--)
      count = count + 1;
  return count;
}
```

Algorithm 13

```
int n, rev;
rev = 0;
while (n > 0) {
  rev = rev * 10 + n % 10;
  n = n / 10;
}
```

Algorithm 14

```
int unknown(int n) {
  int i, j, k = 0;
  for (i = n / 2; i <= n; i++)
    for (j = 2; j <= n; j = j * 2)
      k = k + n / 2;
  return k;
}
```

Algorithm 15

```
for (i = n, j = 0; i >0; i /= 2, j += i);
```

Algorithm 16

```
int fun1(int n) {
  int i, j, k, p, q = 0;
  for (i = 1; i < n; ++i) {
    p = 0;
    for (j = n; j > 1; j = j / 2)
      ++p;
    for (k = 1; k < p; k = k * 2)
      ++q;
  }
  return q;
}
```

Algorithm 17

```
void fun() {
  int i, j;
  for (i = 1; i <= n; i++)
    for (j = 1; j <= log(i); j++)
      printf("GeeksforGeeks");
}
```

Algorithm 18

```
int i = 0, j = 0, y = 0, s = 0;
for (j = 0; j < n + 1; j++)
  y = y + j;
for (i = 1; i <= y; i++)
  s++;
```

Algorithm 19

```
int i = 1, z = 0;
while (z < n * (n + 1) / 2) {
  z += i;
  i++;
}
```