

آنالیز پیچیدگی زمانی تابع های پروژه

بهنوش بهیانی ۹۹۳۶۱۳۰۱۲

main :

```
while{
```

(در صورت وجود فاصله کارکتر هارا شیفت میدهد.)

```
}
```

```
while{
```

(تبدیل آرایه کاراکتر به آرایه ی استرینگ.)

```
}
```

```
while{
```

(تا دفعات بی شمار از کاربر میتواند دستور بگیرد)

```
}
```

پیچیدگی : $3O(n) \rightarrow O(n)$

isOperators :

```
while{
```

(۶ مرتبه برای چک کردن عملگر بودن استرینگ)

```
}
```

پیچیدگی : $O(6)$

IsNumber :

پیچیدگی : $O(1)$

IsError :

```
while{  
    }
```

پیچیدگی : $O(n)$

control_error :

پیچیدگی : $O(1)$

Postfix_To_Infix :

```
while{  
    }
```

پیچیدگی : $O(n)$

insert_tree :

```
while{  
}
```

پیچیدگی : $O(n)$

solve :

```
while{  
    while{
```

(بعد از محاسبه ، جواب را در آرایه ی استرینگ قرار داده و کل اعضای آرایه را ۲ خانه به

چپ شیف می‌دهد)

```
}
```

Postfix_To_Infix:

```
}
```

پیچیدگی : $O(n^2) + O(n^2) = 2O(n^2) \rightarrow O(n^2)$

precedence :

پیچیدگی : $O(1)$

Infix_To_Postfix :

```
while{  
}
```

پیچیدگی : $O(n)$

convert_charTostring :

```
while{  
}
```

پیچیدگی : $O(n)$

تمام متد های درون کلاس ها دارای پیچیدگی $O(1)$ هستند.

تنها متد printBT دارای پیچیدگی $O(n)$ است.