به نام خدا

مستند پروژه پایانی درس کامپایلر

دکتر پارسا

پاییز 1402

اعضای گروه :

علیرضا اسلامی خواه 99521064

مهدی قضاوی 99522014

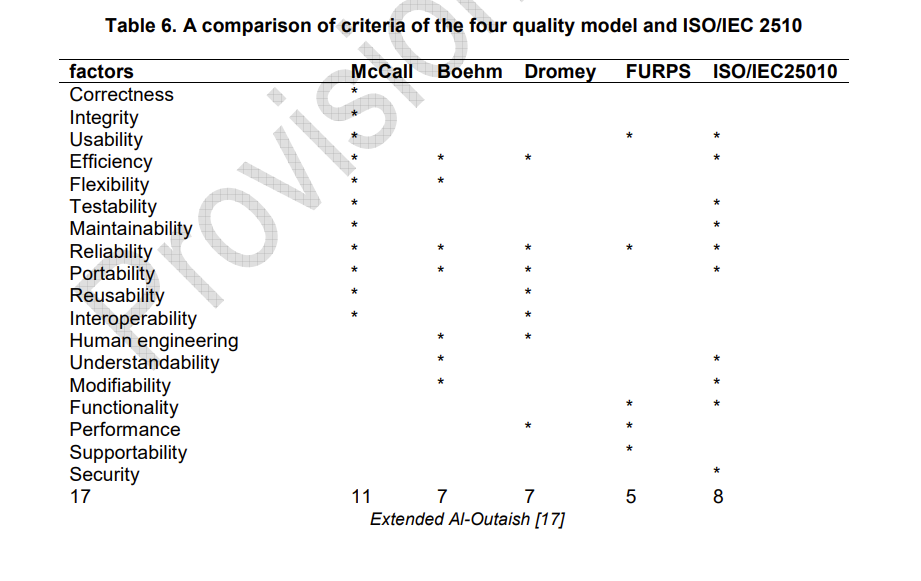
علی سلطانی 99521343

احمدرضا طهماسبی 99521424

سبا رضی 99521316

وحید محمدی 99522077

در مرحله اول با توجه به مقاله باید پیدا میشد که چه چیزهایی میتوانند فاکتورهای سنجش کیفیت یک کد نرم افزاری باشند. با کمی جست و جو در مقاله داده شده در فاز اول به این تیجه رسیدیم که در صفحه 5 مقاله میتوان به این فاکتورها دسترسی پیدا کرد.



سپس به دنبال راه های ارزیابی این ویژگی ها برای سنجش دقیق نرم افزار رفتیم و متوجه شدیم که عمده پارامتر های استفاده شده در این سنجش ها از طریق متریک هایمختلف قابل اندازه گیری است پس اول اینها را پیاده سازی کردیم.

**بخش های مختلف پروژه :**

بخش پیاده سازی شده توسط **علیرضا اسلامی خواه** :

Halstead :

متریک هالستد یک مجموعه از معیارهاست که توسط موریس هالستد (Maurice Halstead) در دهه 1970 برای اندازه‌گیری پیچیدگی کد منبع برنامه نویسی ارائه شد. این معیارها به منظور ارزیابی جنبه‌های مختلف پیچیدگی کد و هزینه‌های پیاده‌سازی برنامه استفاده می‌شوند.

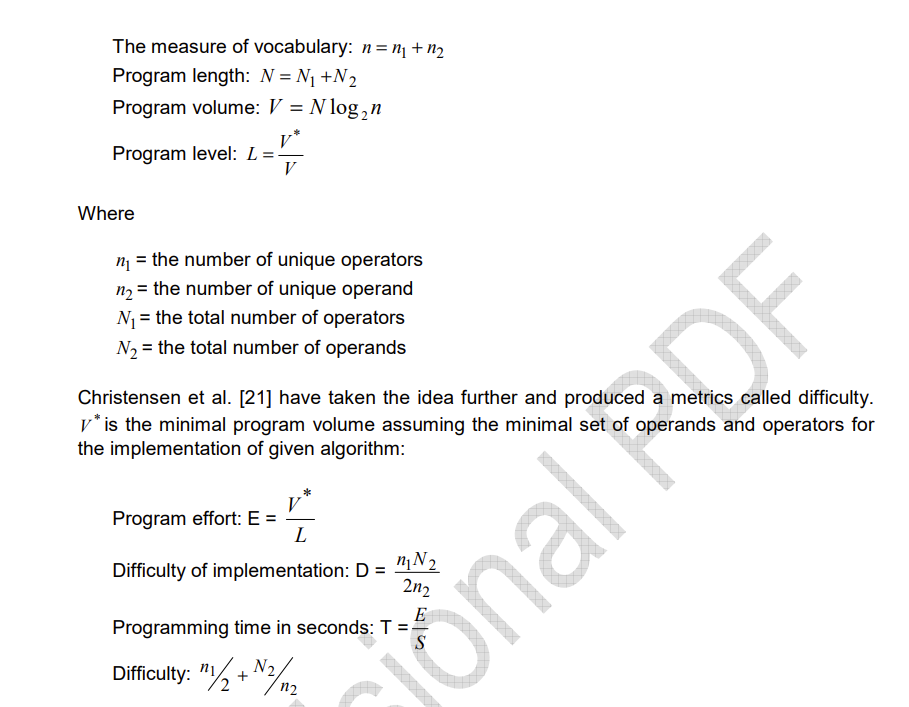
معیارهای هالستد بر مبنای تعداد اجزای مختلف کد منبع محاسبه می‌شوند. این اجزا به دو دسته تقسیم می‌شوند: اجزاي حاصل‌ضرب (Operands) و عملگرها (Operators).

1. عملگرها (Operators): عبارت‌هایی هستند که انجام یک عملیات را نشان می‌دهند. مثال‌هايی از عملگرها عبارتند از `+`, `-`, `\*`, `/` و غیره.

2. اجزاي حاصل‌ضرب (Operands): مقادیری هستند که تحت عملیات‌های انجام شده تغییر می‌کنند. مثال‌هايی از اجزاي حاصل‌ضرب می‌توانند متغیرها، ثابت‌ها و مقادیر متغیرها باشند.

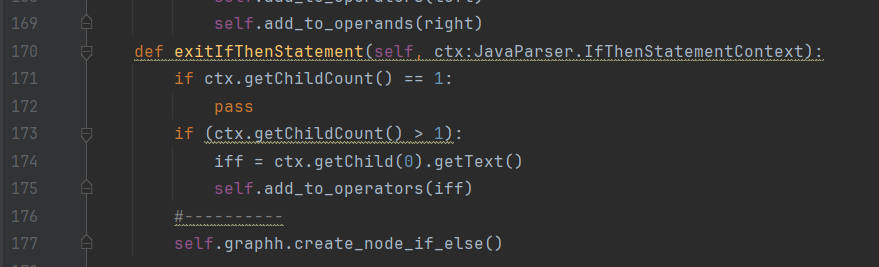
معیار اندازه گیری اپراتورها و اپرندها بر اساس درخت اولیه parser میباشد که با توجه به آن جدول هارا پر میکنیم. مثلا در این شکل بچه اول را اپراتور در نظر گرفته و بچه سوم و چهارم را اپرند.



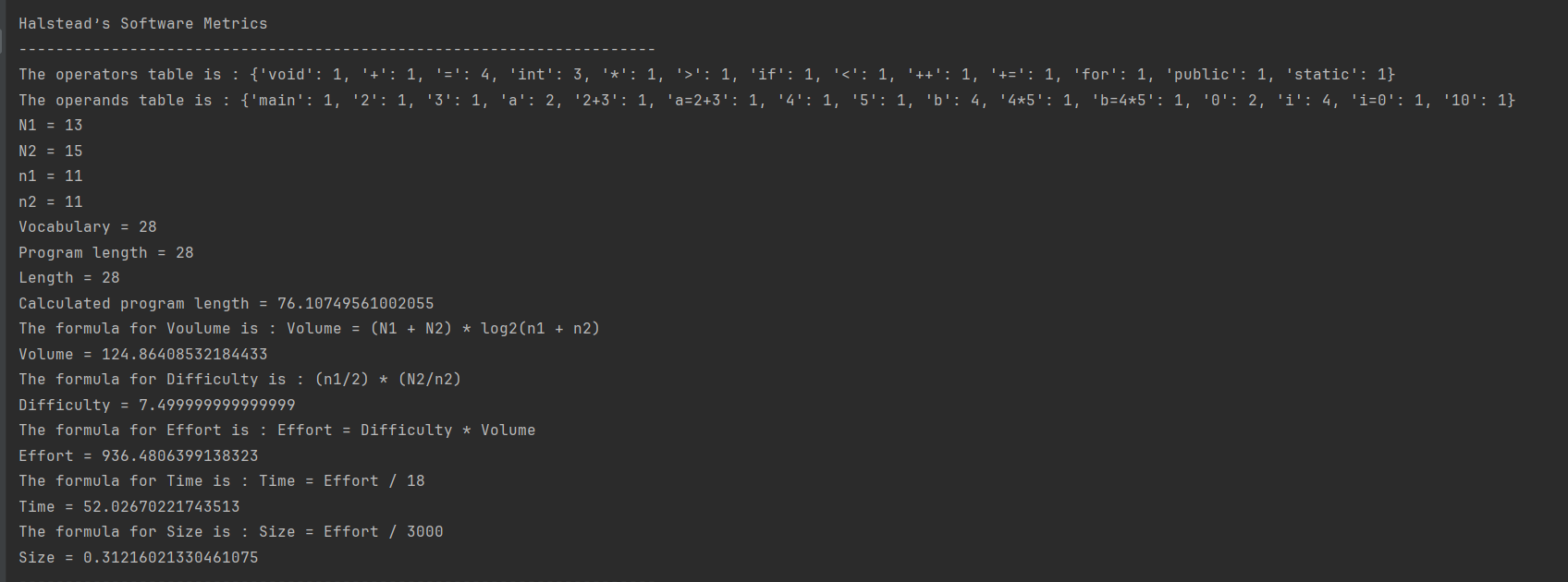


سپس آنها را با توجه به ویژگی که میخواهیم تعریف کنیم استفاده میکنیم.

به عنوان مثال قطعه کد زیر نشان دهنده اینست که چگونه برای ساختار شرط operand ها و operator ها را جدا کرده ایم.



و در آخر هم خروجی دلخواه را گرفتیم:



در اینجا دو جدول operator و operands داریم هر اپراتور یا اپرند رو با توجه به فرکانس تکرار آن نشان میدهد و در آخر مقادیر unique یا یکتا را از آنها استخراج میکند و سپس در فرمول های مختلف میگذارد.

در مرحله آخر باید دو ویژگی flexibility و complexity از کد بررسی میشد.

که متریک های آنها بدین صورت است :

COMPLEXITY :

Cyclomatic Complexity Metric : CC = E – N + 2P

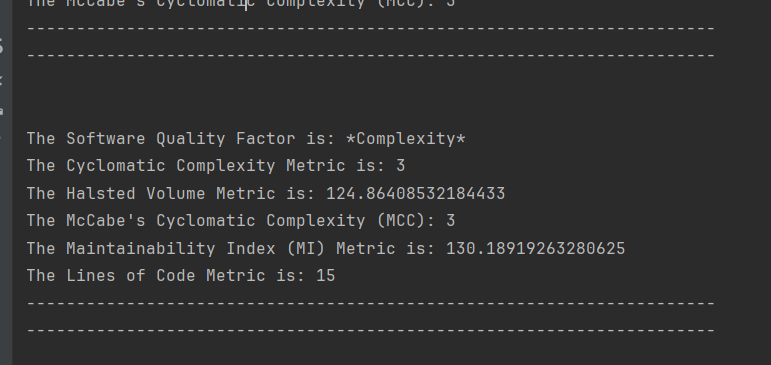
Halsted Volume Metric : HV = N/2\* LOG2(N/L)\* L/V

McCabe's Cyclomatic Complexity (MCC): MCC = E-N+ 2

Maintainability Index (MI) Metric = MI = 171 -5.2LOG2(HV)-0.23\*CC -16.2LOG2(LOC)

Lines of Code Metric: ZIGMA LOCi

خروجی نهایی :



Flexibility :

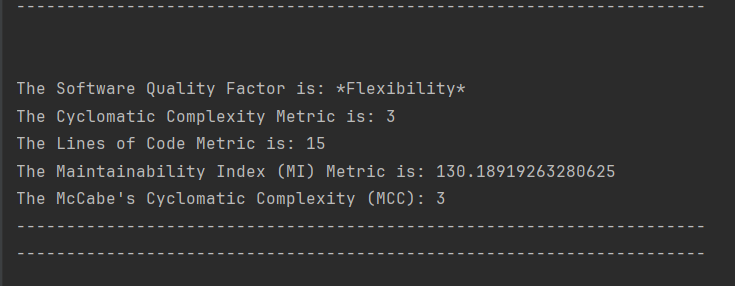
Cyclomatic Complexity Metric : CC = E – N + 2P

Lines of Code Metric: ZIGMA LOCi

Maintainability Index (MI) Metric = MI = 171 -5.2LOG2(HV)-0.23\*CC -16.2LOG2(LOC)

McCabe's Cyclomatic Complexity (MCC): MCC = E-N+ 2

خروجی نهایی :



منابع :

McCabe, T. J. (1976). "A Complexity Measure". In Proceedings of the 2nd International Conference on Software Engineering (ICSE '76).

Microsoft. (2005). "How to: Calculate Maintainability Index." MSDN - Maintainability Index. Microsoft. (2005). "How to: Calculate Maintainability Index." MSDN - Maintainability Index.

Chidamber, S. R., & Kemerer, C. F. (1994). "A Metrics Suite for Object-Oriented Design". IEEE Transactions on Software Engineering, 20(6), 476–493.