

# ソースコード品質解析及び評価ツール

## AdLint オープンソースのC言語向けソースコード静的解析ツールです。

機能① ソースコードに潜在する問題を検出し警告として出力

サンプルソースファイル(sample.c)

```

1 #include <math.h>
2 #include <stdio.h>
3
4 static int CalledRecurisvely(void);
5 static int Function(void) {
6     return CalledRecurisvely();
7 }
8
9 int result = Function() * param;
10 *param = &result;
11 }
12
13 static int CalledRecurisvely(void) {
14     int result;
15     OutputByParam(&result);
16     return result;
17 }
18
19 int main(int argc, char *argv[]) {
20     int i;
21     int npt = NULL;
22     if (argc < 2) {
23         return npt;
24     }
25     npt = CalledRecurisvely();
26 }
27 *npt = 2;
28 return 0;

```

AdLintが出力する警告メッセージ(一部)

ファイル名	行番号	桁番号	ID	メッセージ本文
sample.c	1	1	W0070	このヘッダファイルの内容はソース中で参照されていません。
sample.c	10	5	W0102	仮引数を用いて関数内のローカルな変数のアドレスが返却されています。
sample.c	12	12	W0555	この関数はプロジェクトの他の関数を介して間接的に再帰呼び出しされています。
sample.c	21	16	W0421	値が明らかにNULLとなるポインタを間接参照しています。
sample.c	27	5	W0422	値がNULLかもしれないポインタを間接参照しています。

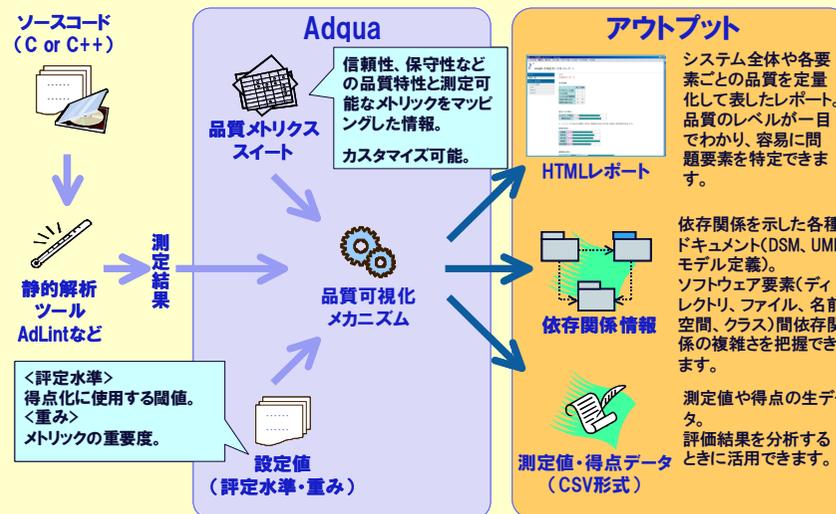
機能② ソースコード品質を把握するためのメトリクスを測定

ソースコードを様々な視点から調査して、多くの品質メトリクスを測定することができます。開発者は「AdLint」が測定する品質メトリクスを定期的に監視することにより、ソフトウェア品質の変化を定量的に評価することができます。

AdLintが測定する品質メトリクス(一部)

経路複雑度	該当関数内に想定される実行経路の複雑さを表す指数
被呼び出し数	該当関数がプロジェクト内の他の関数から呼び出される箇所数
デッドコード数	該当関数内で絶対に実行されない文の数

## Adqua 組込みソフトウェアの品質を定量的に評価できる無償ツールです。



AdLintの仕様

- 対象解析プログラム言語: C言語(ANSI C89, ISO C90, 一部GCCなどの拡張機能)に対応
- 出力される警告メッセージの種類: 720種類
- 測定する品質メトリクスの種類: 15種類
- 出力するコード情報の種類: 14種類
- 動作環境: OS: Windows XP, Windows 7, 各種 Linux ベース OS など、Ruby バージョン 1.9.3-p0 以降が動作するもの
- 動作要件: Ruby バージョン 1.9.3-p0 以降(必須)、GNU Make バージョン3以降(推奨)

Adquaの諸元

- 対象解析プログラム言語: C言語(ANSI C89, ISO C90, 一部GCCなどの拡張機能)に対応
- 対応言語: C言語、C++言語
- 必要なソフトウェア: C言語の場合 AdLint またはQAC(V8.2J~7.2.3J FLEXim 版)、C++言語の場合 QAC++(V2.1J~2.5J FLEXim 版)
- 動作要件: OS: Windows XP Professional (他のWindows OSは動作未確認)、メモリ: 1GB(2GB以上推奨)
- その他制限: 提供先は製造メーカーの組み込みソフトウェア開発現場のみに限らせて頂きます。

両者を連携して使う

