

Agile+ Relief C/C++ 強化ポイント説明書

1. 最新バージョンの強化ポイント (Agile+ Relief V1.1の不具合修正および対応OS追加)

1) 最新OS/MW環境への対応

Agile+ Relief V1.1 C/C++が動作するOS/MW環境として以下を追加しました。

[OS環境]

- Windows (R) 11
- Windows Server (R) 2022
- Red Hat Enterprise Linux 8

[Microsoft (R) Office Excel (R)] ※ 診断レポート生成ツール利用時

- Microsoft (R) Office Excel (R) 2021
- Microsoft (R) Excel (R) for Office 365 バージョン2202にて動作確認済)

[サーバー仮想化ソフトウェア]

- Microsoft (R) Windows Server (R) 2022 Hyper-V (R)
- VMware vSphere 7

2. 過去バージョンの強化ポイント

2. 1. 強化ポイント (2019 → Agile+ Relief V1.1)

- 1) 確認済指摘メッセージの表示抑止機能の追加
過去に検出された指摘と新規指摘に分けて確認することが可能です。
- 2) 指摘の改善 (pgr0431、pgr0520)
- 3) 最新OS/MW環境への対応

2. 2. 強化ポイント (2018autumn → 2019)

- 1) 指摘メッセージを22個追加 (うち10個はCERTオプション用)
- 2) pgr2216の指摘メッセージ改善
- 3) IPA/SEC-C V3 (ESCR-C V3.0) コーディング作法の検査
- 4) CERT C 定義体の追加 (C_20190708_pgr2019)
- 5) 最新OS/MW環境への対応
- 6) pgrfakeコマンドの利便性改善
- 7) CIツール連携 Jenkinsプラグインの提供方法変更

2. 3. 強化ポイント (2018 → 2018autumn)

- 1) 指摘メッセージを4個改善 (すべてCERTオプション用)
- 2) GUI 動作環境の変更 (AdoptOpenJDK 8)
- 3) CERT C 定義体の追加 (C_20180725_pgr2018)

2. 4. 強化ポイント (2017autumn → 2018)

- 1) CERT Cコーディングスタンダードの適合性チェックを追加
↑ CERTオプションの購入が必要
- 2) 指摘メッセージを16個追加 (うち15個はCERTオプション用)
- 3) Visual C++ 2015 の資産に対応

2. 5. 強化ポイント (2017 → 2017autumn)

- 1) IAR Embedded Workbench for ARM 7.4/8.1の資産に対応
- 2) 64ビットOSに対応

- ・ PGRelief C/C++ のすべての機能を 64 ビットOSに対応

2. 6. 強化ポイント (2016autumn → 2017)

- 1) 指摘メッセージを25個追加
- 2) MISRA-C:2012コーディング作法のルール対応率向上
 - ・ すべてのルールに対応 (156ルール中、156ルールに対応)

↑ MISRAオプションの購入が必要
- 3) Visual C++ 2012 / 2013 の資産に対応
- 4) Windows Server (R) 2016 に対応 (フローティング版のみ)
- 5) Microsoft (R) Windows Server (R) 2016 Hyper-V (R) に対応 (フローティング版のみ)
- 6) pgrfakeコマンドの操作性/利便性改善
 - ・ コマンドパラメータ変換定義ファイルの指定方式を追加
 - ・ コンパイラのコンパイル抑止オプションに対応
 - ・ 結果出力先の指定方式を改善

2. 7. 強化ポイント (2016 → 2016autumn)

- 1) SECコーディング作法 (ESCR C++ V2) 適合性チェックを追加
 - ・ 200ルール中、156ルールに対応
- 2) MISRA-C適合性チェックを強化
 - ・ MISRA-C:2012 に MISRA-C:2012 Amendment 1 で追加されたルールを追加
156ルール中、130ルールに対応

↑ MISRAオプションの購入が必要

2. 8. 強化ポイント (2015autumn → 2016)

- 1) セキュリティ脆弱性に関する指摘メッセージを10個追加
- 2) GNU 4.0 ~ 4.8 の C11構文に対応
- 3) Red Hat (R) Enterprise Linux (R) 7 に対応 (フローティング版のみ)
- 4) CI ツール連携 Jenkins プラグインを提供 (フローティング版のみ)
- 5) Visual Studio 2010 アドインを提供 (詳細はホームページ参照)

2. 9. 強化ポイント (2015 → 2015autumn)

- 1) Windows (R) 10 に対応
- 2) サーバー仮想化ソフトウェアに対応 (フローティング版のみ)
 - ・ Microsoft (R) Windows Server (R) 2012 R2 Hyper-V (R) に対応
 - ・ VMware vSphere (R) 5.0 に対応
- 3) GUI使用環境を Java SE Runtime Environment 8 にアップデート

2. 10. 強化ポイント (2014autumn → 2015)

- 1) 指摘メッセージを11個追加
- 2) SECコーディング作法 (ESCR C V2) 適合性チェックを追加
 - ・ 132ルール中、114ルールに対応

2. 11. 強化ポイント (2014 → 2014autumn)

- 1) pgr5コマンドに --includeオプションを追加
→ チェック対象となるソースファイルの解析時に必要な情報を、
解析前に先読みする機能を追加

2. 12. 強化ポイント (2013autumn → 2014)

- 1) gcc4 の資産に対応
 - ・ -K GNU/GCC4 GNU Compiler Collection が提供するC、C++ Compiler 4.0~4.8の予約語に対応
 - ・ -K GNU/GCC4_C90 GNU Compiler Collection が提供するC、C++ Compiler 4.0~4.8の予約語に対応
- 2) コマンドマルチ実行機能の追加
 - ・ 同一端末上において、C/C++解析ライセンスでpgr5、pgrmisra、pgrsecの各コマンドを
4つまで同時に実行させることができる機能を追加

- 3) 64ビット版pgr5、pgrmetricsコマンドの追加
 - ・ pgr5、pgrmetrics コマンドを64ビットOS上で、ネイティブアプリケーションとして動作させるためのモジュールを追加
- 4) 品質監視オプション診断レポート生成ツールの追加
- 5) Windows(R) 8.1 に対応
- 6) Windows Server(R) 2012 R2 に対応

2. 1 3. 強化ポイント (2013 → 2013autumn)

- 1) MISRA-C適合性チェックを強化
 - ・ MISRA-C : 2012 143ルール中、120ルールに対応
- ↑ MISRAオプションの購入が必要

2. 1 4. 強化ポイント (2012autumn → 2013)

- 1) エビデンス出力機能を追加
 - 解析結果をエビデンスとして出力
- 2) 診断レポート生成ツールを追加
 - 解析結果からソースコードの診断レポートを生成 (Excel(R) ブック形式)
- 3) Visual C++ 2010 の資産に対応
- 4) 指摘メッセージを9個追加
- 5) Windows(R) 8 に対応
- 6) Windows Server(R) 2012 に対応

2. 1 5. 強化ポイント (2012 → 2012autumn)

- 1) 対応コンパイラを追加
 - RENESAS/78KOR、RENESAS/RXの予約語に対応

2. 1 6. 強化ポイント (2011autumn → 2012)

- 1) 解析結果集約モードをサポート
 - pgr5、pgrmisra、pgrsecコマンドの解析結果を品質監視オプションで集約
- 2) 指摘メッセージを5個追加

2. 1 7. 強化ポイント (2011 → 2011autumn)

- 1) 指摘メッセージを1個追加

2. 1 8. 強化ポイント (2010autumn → 2011)

- 1) SOFTUNE(R) Development Tools for Eclipse 連携をサポート
(詳細はホームページ参照)
- 2) 指摘メッセージを16個追加
- 3) Windows Server(R) 2008 R2 に対応

2. 1 9. 強化ポイント (2010 → 2010autumn)

- 1) SECコーディング作法 (ESCR C++) 適合性チェックを追加
 - ・ 167ルール中、130ルールに対応
- 2) Windows(R) 7 に対応
- 3) Visual Studio 2008 アドインを提供 (詳細はホームページ参照)

2. 2 0. 強化ポイント (2009autumn → 2010)

- 1) 広域探索機能をサポート (Cソースのみ)
 - ・ 関数の呼び出し経路を追跡してプログラム欠陥を指摘
- ↑ 広域探索オプションの購入が必要
- 2) 指摘メッセージを11個追加 (うち4個は広域探索オプション用)
- 3) Visual Studio 2005 アドインを提供 (詳細はホームページ参照)

2. 2 1. 強化ポイント (2009 → 2009autumn)

- 1) Visual C++ 2008 の資産に対応
- 2) 指摘メッセージを7個追加
- 3) 構成管理ツールとの連携を強化 (品質監視オプション)
→ Interstage (R) Application Development Cycle Manager と連携

2. 2.2. 強化ポイント (2008autumn → 2009)

- 1) 品質分析機能を提供
 - ・ 検出指摘を「品質特性」「問題特性」「文法特性」の観点で分類↑ 品質監視オプションの購入が必要
- 2) 新コマンド (pgrfake) を提供
→ make ファイルに組み込むことで、環境設定を自動抽出・自動解析
- 3) 指摘メッセージビューア (GUI) を改善
 - ・ コマンド解析結果の読み込みをサポート
 - ・ 1FLあたりの同時起動可能数を従来の3倍に拡張
- 4) 解析時間を大幅に短縮
→ プリコンパイルヘッダ機能の拡張や、検査指摘設定により、従来の5倍の処理速度で解析を実行
- 5) 指摘メッセージを9個追加

2. 2.3. 強化ポイント (2008 → 2008autumn)

- 1) MISRA-C++適合性チェックを追加
 - ・ MISRA-C++:2008 228ルール中、164ルールに対応↑ MISRAオプションの購入が必要
- 2) 指摘メッセージを27個追加

2. 2.4. 強化ポイント (2007autumn → 2008)

- 1) 指摘メッセージを29個追加
- 2) 検査指摘設定機能を強化
→ あらかじめ設定したグループや番号の指摘メッセージを出力しない機能を追加
- 3) UTF-8 に対応

2. 2.5. 強化ポイント (2007 → 2007autumn)

- 1) Windows Vista (R) に対応
- 2) Visual C++ 2005 の資産に対応

2. 2.6. 強化ポイント (V3.0a → 2007)

- 1) 組織的な品質管理を強力に支援
 - ・ 構成管理ツールに登録されたソースを一括解析
 - ・ 解析結果を開発者がビューアで確認、対処を記録
 - ・ 開発プロジェクト全体の品質状況が見える化↑ 品質監視オプションの購入が必要
- 2) 変数・関数の命名規約を検査
- 3) SECコーディング作法 (ESCR C) 適合性チェックを強化
 - ・ 129ルール中、114ルールに対応
- 4) MISRA-C適合性チェックを強化
 - ・ MISRA-C : 1998 127ルール中、120ルールに対応
 - ・ MISRA-C : 2004 141ルール中、131ルールに対応↑ MISRAオプションの購入が必要
- 5) 指摘メッセージを22個追加
- 6) Visual Studio .NET 2002/2003 の資産に対応
- 7) Red Hat (R) Linux (R) 9 に対応
↑ フローティング版、コマンド解析のみ
- 8) 英語版をサポート

- メニュー、メッセージ、ドキュメント類を完全英語化
(インストール時に選択)
- 中文フォルダ、中文ソースコード (GB2312) に対応

2. 2 7. 強化ポイント (V3.0 → V3.0a)

- 1) SECコーディング作法 (ESCR C) 適合性チェックを追加
 - ・ 129ルール中、110ルールに対応

2. 2 8. 強化ポイント (V2.0 → V3.0)

- 1) 組織的な品質の管理・把握を容易に実現
 - ↑ 品質監視オプションの購入が必要
- 2) 検査指摘設定機能の提供
 - あらかじめ設定したグループや番号の指摘メッセージだけを出力
- 3) 英語モードをサポート
 - メニュー、メッセージ、ドキュメント類を英語表示
(インストール時に選択)

2. 2 9. 強化ポイント (V1.0 → V2.0)

- 1) MISRA-C適合性チェックを追加
 - ・ MISRA-C : 1998 127ルール中、120ルールに対応
 - ・ MISRA-C : 2004 141ルール中、130ルールに対応
 - ↑ MISRAオプションの購入が必要
- 2) ANSI/C99構文に対応
- 3) 指摘メッセージを133個追加
- 4) 指摘メッセージ表示ウィンドウの処理改善
 - 選択したグループのみで25000件以内であれば表示可
- 5) C++解析時の必要ディスク容量削減 (V1の約40%削減)

3. 指摘メッセージ数

(単位：個)

PGRelief バージョン	C指摘							計
	障害の可能性	移植性・保守性・ 可読性の向上	移植性・保守性・ 可読性の向上 (特にファーム開 発)	堅牢性(セキュリ ティ関連)の向上	効率性の 向上	命名規約違反	その他 (構文の誤りなど)	
V1	102	186	24	0	5	0	42	359
V2-V3	115	218	92	20	5	0	42	492
2007	124	221	96	20	5	5	42	513
2008	127	223	96	20	5	5	42	518
2009	135	229	98	20	5	5	42	534
2010	143	232	98	20	5	5	42	545
2011	147	233	100	20	5	5	42	552
2012	151	234	100	20	5	5	42	557
2013-2014	153	236	105	20	5	5	42	566
2015	153	238	106	28	5	5	42	577
2016	153	238	106	38	5	5	42	587
2017	161	255	106	38	5	5	42	612
2018	161	258	106	51	5	5	42	628
2019	162	262	111	63	5	5	42	650

(単位：個)

PGRelief バージョン	C++指摘							計
	障害の可能性	移植性・保守性・ 可読性の向上	移植性・保守性・ 可読性の向上 (特にファーム開 発)	堅牢性(セキュリ ティ関連)の向上	効率性の 向上	命名規約違反	その他 (構文の誤りなど)	
V1	76	143	16	0	5	0	48	288
V2-V3	86	156	32	20	5	0	46	345
2007	94	159	35	20	5	5	46	364
2008	121	193	78	20	8	5	46	471
2009	128	198	80	20	8	5	46	485
2010	128	201	80	20	8	5	46	488
2011	133	211	88	20	8	5	46	511
2012	137	212	88	20	8	5	46	516
2013-2014	139	214	90	20	8	5	46	522
2015	139	214	90	28	8	5	46	530
2016-2017	139	214	90	37	8	5	46	539
2018	139	217	90	50	8	5	46	555
2019	140	221	93	57	8	5	46	570

4. 強化ポイント詳細(過去10年間分)

各バージョンのリリース時に、どのような改善や変更が行われたか、詳細について説明します。

4. 1. 2019→Agile+ Relief V1.1 強化ポイント詳細

1) 確認済指摘メッセージの表示抑止機能の追加

過去に検出された指摘と新規指摘に分けて確認することが可能です。

-コマンド利用・・・pgr5コマンドに--base、--diffオプションを追加

詳細は、『コマンド解説書』の「1.1.3. オプション説明」を参照してください。

-GUI利用・・・メニューに指摘メッセージ抑止機能を追加

詳細は、Agile+ Reliefのヘルプの「解析オプションを設定する」、「指摘メッセージを見る」を参照してください。

2) 指摘の改善 (pgr0431、pgr0520)

・ pgr0431の改善

C++ソースファイルの解析が中断した際、中断が発生した行を特定しやすくするようにメッセージ内容を見直しました。
詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr0431を参照してください。

・ pgr0520の改善

NULLを含むポインタを返す可能性のある関数(識別子ファイルの [NULL_RETURN_FUNCTION]ラベル配下に登録された識別子と同名の関数)の復帰値の型が不明な場合も、指摘できるようになりました。

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr0520を参照してください。

3) 最新OS/MW環境への対応

[Microsoft (R) Office Excel (R)] ※ 診断レポート生成ツール利用時

- Microsoft (R) Excel (R) for Office 365

4. 2. 2018autumn→2019 強化ポイント詳細

4. 2. 1. 指摘の追加

次の22個の指摘を追加しました。

ID	指摘メッセージ	Gr	C/C++	備考
pgr1242	オペランドに @1 を使用する式 @2 は不要な演算を含む可能性があります。	b	C/C++	
pgr1243	代入 @1 の結果を他の式 @2 で使用しています。	g	C/C++	
pgr1250	ポインタ型に '¥0' を使用しています。	b	C/C++	
pgr1251	char型に (void*)0 を使用しています。	b	C/C++	
pgr1252	char型に 0 を使用しています。	g	C/C++	
pgr1253	配列 @1 の初期化指示子 @2 は、配列の範囲を超えています。	a	C/C++	
pgr1254	初期化指示子を使用している配列の要素数が省略されています。	b	C/C++	
pgr1255	初期化指示子を使用している配列の終端を初期化していません。	g	C/C++	
pgr1256	マルチスレッド関数内 @1 でビットフィールド @2 が排他制御されていません。	h	C/C++	
pgr1257	マルチスレッド関数内 @1 でビットフィールド @2 が排他制御されていません。	h	C/C++	
pgr1807	型 @1 を使用していません。	g	C	
pgr1808	タグ @1 を使用していません。	g	C	
pgr6016	プログラム終了ハンドラ @1 で、関数 @2 を呼び出した場合、未定義の動作を引き起こす可能性があります。(終了ハンドラ登録 : @3 の @4 行目)	h	C	CERT
pgr6020	関数 @1 に渡している書式指定引数 @2 が外部の情報を使う場合、外部からの攻撃に利用される危険性があります。	h	C/C++	CERT
pgr6021	関数 @1 の結果を @2 型に代入すると、ファイル終端をEOFやWEOFで正しく判断できない可能性があります。関数と同じ @3 型にしたほうが良いでしょう。	h	C/C++	CERT
pgr6022	文字入力関数 @1 の戻り値を確認しただけではストリーム @2 の状態を正しく判断できない可能性があります。ストリームの状態を確認するには関数feofやferrorなどで確認したほうが良いでしょう。	h	C/C++	CERT
pgr6030	@1 は @2 の範囲を超えてアクセスしてしまう可能性があります。(領域サイズ : @3, 複写サイズ : @4)	h	C/C++	CERT
pgr6031	@1 は @2 の範囲を超えてアクセスしてしまう可能性があります。(領域サイズ : @3, 複写サイズ : @4)	h	C/C++	CERT
pgr6050	シグナルハンドラ @1 から非同期安全ではない関数 @2 を呼び出しているため、シグナル処理が安全に行われません。(シグナル登録 : @3 の @4 行目)	h	C	CERT
pgr6051	シグナルハンドラ @1 から非同期安全ではない変数 @2 を参照しているため、シグナル処理が安全に行われません。(シグナル登録 : @3 の @4 行目, 変数宣言 : @5 の @6 行目)	h	C	CERT
pgr6052	シグナルハンドラ @1 で呼び出している関数 @2 は、@3 の @4 行目で非同期安全ではない関数 @5 を呼び出しているため、シグナル処理が安全に行われません。(シグナル登録 : @6 の @7 行目)	h	C	CERT
pgr6053	シグナルハンドラ @1 で呼び出している関数 @2 は、@3 の @4 行目で非同期安全ではない変数 @5 を参照しているため、シグナル処理が安全に行われません。(シグナル登録 : @6 の @7 行目, 変数宣言 : @8 の @9 行目)	h	C	CERT

備考欄に記載されているキーワードの意味 :

CERT : CERTオプションのライセンス所有時に抽出される指摘メッセージです。

4. 2. 2. 指摘の改善

pgr2216の指摘の対象を、C++ソースファイルからC/C++ソースファイルへ拡張

インライン関数化やテンプレート関数化が難しいマクロ関数に対する指摘を抑止する識別子ラベルを追加

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr2216を参照

4. 2. 3. IPA/SEC-C V3 (ESCR-C V3.0) コーディング作法の検査

IPA/SEC-C V3 (ESCR-C V3.0) コーディング作法の検査が可能

4. 2. 4. CERT C定義体の追加

CERT C コーディングスタンダード検査時に使用する、以下のバージョンのCERT C 定義体を追加

- ・ C_20190708_pgr2019 を追加

本定義体を使用することで、2018年7月25日時点におけるSEI CERT C Coding Standard

(<https://wiki.sei.cmu.edu/confluence/display/c/SEI+CERT+C+Coding+Standard>)

のコーディングルールに沿って、CERT C コーディングスタンダードの検査が可能

4. 2. 5. 最新OS/MW環境への対応

[OS環境] ※ フローティングライセンス版のみ対応

- Windows Server (R) 2019

[サーバー仮想化ソフトウェア] ※ フローティングライセンス版のみ対応

- Microsoft (R) Windows Server (R) 2019 Hyper-V (R)

[Microsoft (R) Office Excel (R)] ※ 診断レポート生成ツール利用時

- Microsoft (R) Office Excel (R) 2016
- Microsoft (R) Office Excel (R) 2019

4. 2. 6. pgrfakeコマンドの利便性改善

pgrfakeコマンドで解析コマンドの一括実行ができるよう改善

詳細は、『コマンド解説書』の「1.3. pgrfakeコマンド」、

「5.2. pgrfakeコマンドによる解析コマンドの一括実行（全てのお客様へ）」を参照

4. 2. 7. CIツール連携 Jenkinsプラグインの提供方法変更

フローティングライセンス版のインストール媒体に格納し提供していた『CIツール連携 Jenkinsプラグイン』を、個別提供に変更

『CIツール連携 Jenkinsプラグイン』を必要な方はお問い合わせください。

※ 『CIツール連携 Jenkinsプラグイン』で利用するオープンソースソフトウェアにEOL (End-of-life) やセキュリティ脆弱性等の問題が発生した場合、随時対応できるよう提供方式を変更

4. 3. 2018→2018autumn 強化ポイント詳細

4. 3. 1. 指摘の改善

- ・ pgr6000、pgr6005 の改善

ラップした関数で検出される指摘を抑止する識別子ラベルを追加しました。

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr6000、6005を参照してください。

- ・ pgr6001 の改善

チェック対象となるミューテックスが初期化されている場合、指摘しないよう改善しました。

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr6001を参照してください。

- ・ pgr6002 の改善

ポインタ型引数への直前の代入が外部変数の場合、指摘しないよう改善しました。

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr6002を参照してください。。

4. 3. 2. GUI 動作環境の変更

GUIが使用するJavaの環境を、AdoptOpenJDK 8 に変更しました。

4. 3. 3. CERT C 定義体の追加

CERT C コーディングスタンダード検査時に使用する、以下のバージョンのCERT C 定義体を追加
 義体を追加しました。

- ・ C_20180725_pgr2018

本定義体を使用することで、2018年7月25日時点におけるSEI CERT C Coding Standard
 (<https://wiki.sei.cmu.edu/confluence/display/c/SEI+CERT+C+Coding+Standard>)
 のコーディングルールに沿って、CERT C コーディングスタンダードの検査が可能です。

4. 4. 2017autumn→2018 強化ポイント詳細

4. 4. 1. CERT Cコーディングスタンダードの検査

CERT Cコーディングスタンダードを検査することが可能になりました。

なお、CERT Cコーディングスタンダードを検査する場合には、CERTオプションのライセンスが必要になります。

詳細は、『CERTオプション使用説明書』、および『指摘詳細説明書』の「4. CERTオプション」を参照してください。

4. 4. 2. 指摘の追加

次の16個の指摘を追加しました。

ID	指摘メッセージ	Gr	C/C++	備考
pgr1166	下線で始まる識別子 @1 はANSIで予約済みのため、未定義の動作を引き起こす可能性があります。	b	C/C++	
pgr6000	ロックしていない @1 を 関数 @2 に渡すのは誤りの可能性があります。	h	C/C++	CERT
pgr6001	関数内でロックまたはアンロックしていない @1 を 関数 @2 で破棄しています。他のスレッドで使用するオブジェクトを破棄していないか確認してください。	h	C/C++	CERT
pgr6002	関数 @1 の @2 番目の実引数 @3 は外部変数または静的変数ではないためスレッド実行時に有効ではない可能性があります。	h	C/C++	CERT
pgr6003	@1 型の変数 @2 は外部変数または静的変数ではないためスレッド実行時に有効ではない可能性があります。	h	C/C++	CERT
pgr6004	関数 @1 の順番が異なるため、関数 @2 を並列実行するとデッドロックとなる可能性があります。(比較対象 : @3 行目と @4 行目, 指摘対象 : @5 行目と @6 行目)	h	C/C++	CERT
pgr6005	関数 @1 で ロックした @2 はアンロックしないとデッドロックとなる可能性があります。(ロック回数:@3、アンロック回数:@4)	h	C/C++	CERT
pgr6006	排他制御中に他の処理によってブロックされる可能性のある関数 @1 を呼び出すと、デッドロックとなる可能性があります。	h	C/C++	CERT
pgr6007	浮動小数点型を含むデータの比較を関数 @1 で行うのは好ましくありません。	b	C/C++	CERT
pgr6008	マルチスレッド関数内でスレッドアンセーフな関数 @1 を使用しています。	h	C/C++	CERT
pgr6009	関数mainの仮引数 @1 には @2 の結果は反映されない可能性があります。	b	C/C++	CERT
pgr6010	関数 @1 の @2 番目の引数は @3 によるバイトオーダーの変換を行っていません。	h	C/C++	CERT
pgr6011	関数 @1 の @2 番目の引数に @3 を渡すと、スレッドが非同期でキャンセルされる可能性があります。正しい引数を渡しているか確認してください。	h	C/C++	CERT
pgr6012	@1 行目の関数 @2 と実引数が異なります。(条件変数 : @3, ミューテックス : @4)	h	C/C++	CERT
pgr6013	charが符号ありならば char型の @1 をサイズの大きな整数へ変換の前に unsigned char でキャストしないと、符号拡張が発生しビットパターンが変わる可能性があります。	h	C/C++	CERT
pgr6015	関数 @1 の呼び出しの前で @2 に @3 を設定していません。関数の @1 では @2 の初期化は行いません。呼出の前に @3 を設定してください。	h	C/C++	CERT

備考欄に記載されているキーワードの意味 :

CERT : CERTオプションのライセンス所有時に抽出される指摘メッセージです。

4. 4. 3. 対応するコンパイラの追加

コンパイラ仕様を指定する-Kオプションに、以下を追加しました。

| -K MS/VC2015

Microsoft社製Microsoft Visual C++ 2015 (Win32ビット)の予約語に対応します。

4. 5. 2017→2017autumn 強化ポイント詳細

4. 5. 1. 対応するコンパイラを追加

コンパイラ仕様を指定する-Kオプションに、以下を追加しました。

| -K IAR/EWARM7.4

IARシステムズ社製IAR Embedded Workbench for ARM 7.4 コンパイラの予約語に対応します。

| -K IAR/EWARM8.1

IARシステムズ社製IAR Embedded Workbench for ARM 8.1 コンパイラの予約語に対応します。

4. 5. 2. 64ビットOSへの対応

PGRelief C/C++(64ビット版)を追加しました。PGRelief C/C++のすべての機能が、

64ビットOS上でネイティブアプリケーションとして動作します。

GUIやpgrfakeコマンドで、C/C++言語で書かれた大規模なソースプログラムを解析できるようになりました。

4. 5. 3. pgr1395の改善

大量のメッセージ出力により解析時間が長くなる場合を考慮し、

pgr1395のメッセージ出力数に関する初期設定を変更しました。

[変更前]

初期設定) すべての検出箇所に対してメッセージを出力する。

[変更後]

初期設定) 検出するリテラルの種類ごとに、解析対象ソースファイルで1箇所のみメッセージを出力する。

詳細は、『指摘詳細説明書』の「2. 指摘メッセージ詳細説明」- pgr1395を参照してください。

4. 6. 2016autumn→2017 強化ポイント詳細

4. 6. 1. 指摘の追加

次の25個の指摘を追加しました。

ID	指摘メッセージ	Gr	C/C++	備考
pgr0803	可変長の配列 @1 が宣言されています。	g	C	
pgr0804	マクロ名に使用している @1 は予約語です。	g	C	
pgr0807	型 @1 を使用していません。	g	C	
pgr0808	タグ @1 を使用していません。	g	C	
pgr0809	マクロ @1 を使用していません。	g	C	
pgr0810	//コメントが行連結されています。	g	C	
pgr0813	8進あるいは16進エスケープシーケンスと通常文字が混在しています。	g	C	
pgr0814	外部結合をもつ識別子 @1 の長さが @2 文字を超えました。	b	C	
pgr0815	@1 とファイル @2 の @3 行目の @4 の名前は、大文字と小文字の違いだけです。	b	C	
pgr0816	識別子 @1 の長さが @2 文字を超えました。	g	C	
pgr0817	マクロ名 @1 の長さが @2 文字を超えました。	g	C	
pgr0818	@1 は同じ要素を初期化しています。	b	C	
pgr0819	配列 @1 は要素を個別に初期化していますが、配列サイズを明示的に指定していません。	g	C	
pgr0820	関数 @1 の @2 番目の実引数 @3 は @4 型にした方がよいでしょう。 (実引数の型: @5)	g	C	
pgr0825	@1 は異なる本質型へのキャストです。(キャスト型: @2, 式の型: @3)	g	C	
pgr0826	代入式 @1 は void * 型のポインタを @2 型に代入しています。	g	C	
pgr0827	@1 は、@2 の @3 行目で定義したマクロ関数では置換されません。	b	C	
pgr0828	@1 の初期化子並びの式の中に副作用を発生させる式 @2 があります。	b	C	
pgr0829	関数 @1 の @2 番目の実引数 @3 と、対応する仮引数 @4 (@5 の @6 行目)の配列サイズが異なります。(実引数: @7, 仮引数: @8)	g	C	
pgr0830	関数 @1 の @2 番目の実引数 @3 と、対応する仮引数 @4 (@5 の @6 行目)には配列サイズが指定されていません。(実引数: @7, 仮引数: @8)	g	C	
pgr0848	#ifや#elif文中の整数の記述が0でも1でもありません。	g	C	

pgr0849	コード部をコメントアウトしているかもしれません。	g	C	
pgr1237	文字列の比較を関数 @1 で行うのは好ましくありません。	b	C	
pgr1239	関数 @1 の @2 番目の実引数 @3 と、@4 番目の実引数 @5 の型が異なります。(@2 番目の実引数 : @6 , @4 番目の実引数 : @7)	b	C	
pgr1395	リテラル @1 を使用しています。(リテラルの種類 @2)	g	C	

4. 6. 2. MISRA-C:2012コーディング作法のルール対応率向上

MISRA-C:2012 (MISRA-C:2012 Amendment 1含む) コーディング作法のすべてのルールに対応しました。

詳細は、『MISRAオプション使用説明書』の「3.3 MISRA-C V3」を参照してください。

4. 6. 3. 対応するコンパイラの追加

コンパイラ仕様を指定する-Kオプションに、以下を追加しました。

| -K MS/VC2012

Microsoft社製Microsoft Visual C++ 2012(Win32ビットでMFC未使用)の予約語に対応します。

| -K MS/VC2012MFC

Microsoft社製Microsoft Visual C++ 2012(Win32ビットでMFC使用)の予約語に対応します。

| -K MS/VC2013

Microsoft社製Microsoft Visual C++ 2013(Win32ビットでMFC未使用)の予約語に対応します。

| -K MS/VC2013MFC

Microsoft社製Microsoft Visual C++ 2013(Win32ビットでMFC使用)の予約語に対応します。

4. 6. 4. 最新OS/MW環境への対応

PGRelief C/C++が動作するOS/MW環境として以下を追加しました。

[OS環境] ※ フローティングライセンス版のみ対応

- Windows Server (R) 2016

[サーバー仮想化ソフトウェア] ※ フローティングライセンス版のみ対応

- Microsoft (R) Windows Server (R) 2016 Hyper-V (R)

4. 6. 5. pgrfakeコマンドの操作性/利便性改善

pgrfakeコマンドを以下のとおり改善しました。

- 環境変数“PGRFAKE_INI_FILE”を追加

pgrfakeコマンド実行時に、任意のコマンドパラメタ変換定義ファイルを指定できるようになりました。

- コマンドパラメタ変換定義ファイルに“NOANALYZEKEY”キーを追加

コンパイラのコンパイル抑止オプション(gccの-Mオプション等)を識別し、

PGReliefの解析実行を抑止できるようになりました。

- コマンドパラメタ変換定義ファイルの“OUTPUTDIR”キーの指定方式を改善

pgrfakeコマンドの実行結果出力先ディレクトリの指定に、~(ホームディレクトリ)が指定できるようになりました。

※ Red Hat Enterprise Linux (R)上で動作する場合のみ対応

詳細は、『コマンド解説書』の「1.3. pgrfakeコマンド」を参照してください。

4. 7. 2016→2016autumn 強化ポイント詳細

4. 7. 1. IPA/SEC-C++ V2 (ESCR-C++ V2.0) コーディング作法の検査

IPA/SEC-C++ V2 (ESCR-C++ V2.0) コーディング作法を検査することが可能になりました。

4. 7. 2. MISRA-C:2012 Amendment 1コーディング作法の検査

MISRA-C:2012に加えMISRA-C:2012 Amendment 1で追加されたコーディング作法を検査することが可能となりました。

4. 8. 2015autumn→2016 強化ポイント詳細

4. 8. 1. 指摘の追加

次の10個の指摘を追加しました。

ID	指摘メッセージ	Gr	C/C++
pgr1160	マルチスレッド関数内で変数 @1 が排他制御されていません。	h	C/C++
pgr1161	@1 の @2 行目の関数 @3 の呼び出しから、関数 @4 の呼び出しまでの間に、権限の変更がありません。	h	C

pgr1162	文字列リテラル @1 は機密情報である可能性があります。	h	C/C++
pgr1163	@1 で資源を割り当てた @2 は領域を初期化していません。	h	C/C++
pgr1164	@1 で環境変数の値を @2 にコピーすると、配列の範囲外に書き込みをする可能性があります。	h	C/C++
pgr1170	@1 @2 を使用しています。@2 より @3 の使用を検討しましょう。	h	C/C++
pgr1171	@1 @2 を使用しています。ファイル名に依存する @2 はTOCTOU競合攻撃に対して脆弱です。@3	h	C/C++
pgr1172	@1 @2 を使用しています。@2 はコマンドインジェクションに対し脆弱です。	h	C/C++
pgr1173	@1 @2 を使用しています。@2 はroot権限で実行しているか確認してください。	h	C/C++
pgr1174	@1 @2 を使用しています。root権限を設定した場合、権限が正しく破棄されているか確認してください。	h	C/C++

4. 8. 2. 対応するコンパイラの追加/変更

コンパイラ仕様を指定する-Kオプションに、以下を追加しました。

| -K GNU/GCC4_C11

GNU Compiler Collection が提供するC、C++ Compiler 4.0~4.8の予約語、および、C11構文に対応します。

セキュリティ脆弱性に関する指摘の追加に伴い、以下のコンパイラ仕様の登録識別子（推奨/非推奨関数など）を見直しました。

| -K MS/VC2005
| -K MS/VC2005MFC
| -K MS/VC2008
| -K MS/VC2008MFC
| -K MS/VC2010
| -K MS/VC2010MFC

4. 8. 3. 最新OS環境への対応

PGRelief C/C++が動作するOS環境として以下を追加しました。

[OS環境]

- Red Hat (R) Enterprise Linux (R) 7

※ フローティングライセンスの64ビット版のみ対応

4. 8. 4. CI ツール(Jenkins)のプラグイン提供

CI (Continuous Integration) ツールのJenkinsと連携するプラグインを追加しました。

プラグインを適用することで、PGRelief C/C++の実行結果をJenkins上で集約、表示できるようになりました。

詳細は、インストール媒体に格納された『CIツール連携 Jenkinsプラグイン使用説明書』

("Etc\ci-plugin\pgrelief-jenkins-plugin_jp.pdf") を参照してください。※ フローティングライセンス版のみ対応

4. 9. 2015→2015autumn 強化ポイント詳細

4. 9. 1. 最新OS/MW環境への対応

PGRelief C/C++が動作するOS/MW環境として以下を追加しました。

[OS環境]

- Windows (R) 10

[サーバー仮想化ソフトウェア] ※ フローティングライセンス版のみ対応

- Microsoft (R) Windows Server (R) 2012 R2 Hyper-V (R)

- VMware vSphere (R) 5.0

4. 9. 2. GUI動作環境の更新

GUIが使用するJava SE Runtime Environmentの環境を、Java SE Runtime Environment 8 にアップデートしました。

4. 10. 2014autumn→2015 強化ポイント詳細

4. 10. 1. 指摘の追加

次の11個の指摘を追加しました。

ID	指摘メッセージ	Gr	C/C++
----	---------	----	-------

pgr0856	条件式 @1 はポインタ型と列挙定数を比較しています。(ポインタの型 : @2)	b	C
pgr0857	条件式 @1 はポインタ型と整数定数を比較しています。(ポインタの型 : @2)	b	C
pgr0858	条件式 @1 はポインタ型と整数定数を比較しています。(ポインタの型 : @2)	g	C
pgr1151	関数 @1 を呼び出す前に関数 @2 を呼び出していません。	h	C/C++
pgr1152	関数 @1 の @2 番目の実引数に定数が渡されています。	h	C/C++
pgr1153	プロセスを生成する前にファイルをクローズしていない可能性があります。(ファイルオープン行 : @1 行目)	h	C/C++
pgr1154	関数 @1 で読み取った変数 @2 の長さは0である可能性があります。	h	C/C++
pgr1155	関数 @1 の @2 番目の実引数が指定されていません。	h	C/C++
pgr1156	符号なし整数型の式"@1"は事前に内容を確認していません。演算がラップアラウンドする可能性があります。	h	C/C++
pgr1157	符号あり整数型の式"@1"は事前に内容を確認していません。演算がオーバーフローする可能性があります。	h	C/C++
pgr1158	関数 @1の引数 @2 と @3 は事前に内容を確認していません。引数の乗算結果がラップアラウンドする可能性があります。	h	C/C++

4. 1 0. 2. IPA/SEC-C V2 (ESCR-C V2) コーディング作法の検査

IPA/SEC-C V2 (ESCR-C V2) コーディング作法を検査することが可能になりました。

4. 1 1. 2014→2014autumn 強化ポイント詳細

4. 1 1. 1. pgr5コマンド --includeオプションの追加

pgr5コマンドに--includeオプションを追加しました。

詳細は、『コマンド解説書』の「1.1.3. オプション説明」を参照してください。

4. 1 2. 2013autumn→2014 強化ポイント詳細

4. 1 2. 1. 対応するコンパイラの追加

コンパイラ仕様を指定する-Kオプションに、以下を追加しました。

| -K GNU/GCC4

GNU Compiler Collection が提供するC、C++ Compiler 4.0~4.8の予約語に対応します。

| -K GNU/GCC4_C90

GNU Compiler Collection が提供するC、C++ Compiler 4.0~4.8の予約語

(-ansiや-std=などでC90構文を指定している場合)に対応します。

4. 1 2. 2. コマンドマルチ実行機能の追加

1つの端末上で、C/C++解析ライセンスで、pgr5、pgrmisra、pgrsecコマンドを同時に4つまで実行させることができる機能を追加しました。

詳細は、『コマンド解説書』の「1.1.6 環境変数」を参照してください。

4. 1 2. 3. 64ビット版pgr5、pgrmetricsコマンドの追加

pgr5、pgrmetrics コマンドを64ビットOS上で、ネイティブアプリケーションとして動作させるためのモジュールを追加しました。

4. 1 2. 4. 品質監視オプション診断レポート生成ツールの追加

品質監視オプションの実行結果からソース診断のレポートを生成する診断レポート生成ツールを追加しました。

情報を分析した結果をMicrosoft (R) Office Excel (R)のブック形式のレポートとして出力します。

4. 1 2. 5. 最新OS/MW環境への対応

PGRelief C/C++が動作するOS/MW環境として以下を追加しました。

[OS環境]

- Windows (R) 8.1

- Windows Server (R) 2012 R

- ※ SOFTUNEは富士通セミコンダクター株式会社の登録商標です。
- ※ MISRAとそのロゴは、MISRAコンソーシアムを代表してMIRA Ltd. の登録商標です。
- ※ Red Hat は米国およびその他の国における Red Hat Inc. の登録商標もしくは商標です。
- ※ Linux は Linus Torvalds の商標です。
- ※ CERTは、米国Carnegie Mellon Universityの登録商標です。
- ※ 本資料に記載されている会社名および製品名称は各社の商標または登録商標です。
- ※ 本資料に記載されているシステム名、製品名等には必ずしも商標表示（R）、TM）を付記していません。

株式会社富士通

Copyright 2023 FUJITSU LIMITED