

プログラム初学者へのコンパイルエラー修正支援システム — エラーメッセージを用いた修正支援 —

A System to Correct Compilation Errors for Novice Programmers: System Support for Correcting the Errors by using Error Messages

西村 将広[†]
Masahiro Nishimura[†]

橋浦 弘明[‡]
Hiroaki Hashiura[‡]

古宮 誠一[‡]
Seiichi Komiya[‡]

[†] 芝浦工業大学 工学部

[‡] 芝浦工業大学大学院 工学研究科

[†] Faculty of Engineering, Shibaura Institute of Technology.

[‡] Graduate School of Engineering, Shibaura Institute of Technology.

要旨

今日、大学ではプログラミングに関する授業が行われているが、C言語の演習においてコンパイルエラーを修正できず、つまずいてしまう学生が多い。コンパイルエラーは、修正ができないとプログラムが動作しないため、論理的なエラーの発見やアルゴリズムの実装など、それ以降の学習に支障が出てしまう。コンパイルエラーは、本来ならば、コンパイラから出力されるエラーメッセージをもとに修正するのが望ましいが、現状では指導者の直接指導に頼っているが、それも十分ではない。そのため、本研究では、コンパイル時に出力されるエラーメッセージを利用して、コンパイルエラーの修正を支援するシステムの研究開発を行う。

1. はじめに

今日、大学ではプログラミングに関する授業が行われているが、C言語の演習においてコンパイルエラーを修正できずつまずいてしまう学生が多い。コンパイルエラーは、修正ができないとプログラムが動作しないため、論理的なエラーの発見やアルゴリズムの実装など、それ以降の学習に支障が出てしまう。これらのエラーは、本来ならばコンパイラから出力されるエラーメッセージによって解決されることが望ましいが、数行のエラーメッセージから原因を特定し、対処することは初学者にとって容易ではない。そのため、指導者による直接指導が有効だが、実際の授業では指導者に対して学生の方が多く、全てに対応するのは物理的に不可能である。そこで本研究では、C言語を学習するプログラム初心者に対して、コンパイルエラー修正の助けとなる情報を提供するシステムを提案する。

2. コンパイルエラーの位置づけ

図1にプログラムを正しく動作するようになるまでの大まかな流れを示した。図のようにプログラムが動作するまでには、コンパイルエラーの除去と論理的エラーの除去という二つの壁が存在する。1つは、コンパイラがエラーを出力し、実行可能ファイルを生成できないコンパイルエラーで、セミコロン忘れなど構文的な問題によるエラーである。もう1つが、実行は可能な状況だが、意図どおりに動作しない論理的エラーで、for文の条件設定ミスなど論理的な部分でのエラーである。そのうち、コンパイルエラーの修正はソースコードを書いた後の最初のステップであり、初学者が最初につきあたる壁となる。

また、コンパイルエラーの特徴として、プログラムとして動作しないため、printfデバッグのような手法が使えないことが挙げられる。このため、修正の頼りとなるのは数行のエラーメッセージのみであり、修正のための情報が少ない。

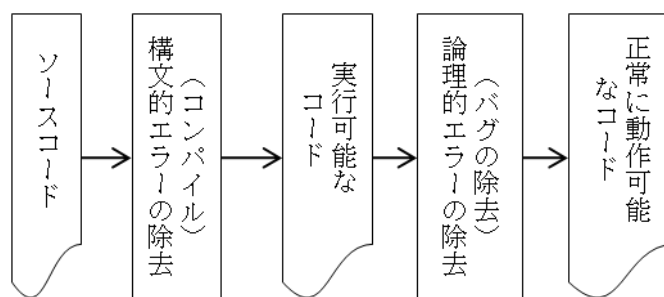


図1 プログラムが正しく動くまでの流れ

3. コンパイルエラーを修正できない原因

学習者がコンパイルエラーを修正できない原因として、以下の三つを考える。

3.1. エラーメッセージの内容を理解できない

一般に、コンパイルエラーは英語で出力される。しかし、エラーメッセージはプログラミング言語独特の表現などを含んでおり、初心者がその意味をとらえることは難しい。"syntax error at end of input"における"input"を例に挙げると、直訳すれば"入力"だが、実際の意味としては"コンパイルしようとしたファイルに記述したソースコード"であり、"input"からこの意味にたどり着くのは難しい。このように、単純な英語知識だけで正しい意味をとらえることは困難である。また、"lvalue"のように、辞書に載っていない単語もあることから、意味を把握することが容易でないとわかる。

3.2. エラーメッセージへの対処方法がわからない

エラーメッセージの意味を正しく理解できた場合でも、プログラムに対しての構文知識の不足から、エラーへの対処方法がわからない場合が考えられる。例えば、"conflicting types for 'test'"というエラーが図2のようなコードで発生する。このエラーは、"関数testが二重に宣言されている"という意味だが、エラーの原因となっているのは、関数main内で利用されている関数testがmainよりも後に宣言されていることであり、エラーメッセージの意味から、この原因を類推することは容易でない。また、このエラーの修正には、"ある関数内で呼び出す関数はそれより前に書かなくてはならない"という文法知識が必要である。このように、言語によって異なる場合もある、様々な文法構文の知識を網羅することも、初学者にとって容易ではないため、これも修正を困難にしている原因と言える。

```
int main() {
    test();
}
void test() {...}
```

図2 エラーを示すコード

3.3. 適切な修正方法を選択できない

図3で示すように、1つのエラーメッセージに対して、発生する状況や対処方法が複数存在しうること修正の妨げになっている。例えば、"syntax error before ')' token" (閉じ括弧の前の文字に問題がある) とエラーが出た場合、ソースコードによって、全角スペースが入っていることや、括弧の中の文にセミコロンをつけ忘れているなど、様々な可能性が考えられるため、修正方法も様々である。このように、エラーメッセージとエラーの除去方法は1対多の関係にあるため、うまく修正方法を見出すには経験や知識が必要であり、初心者には困難である。

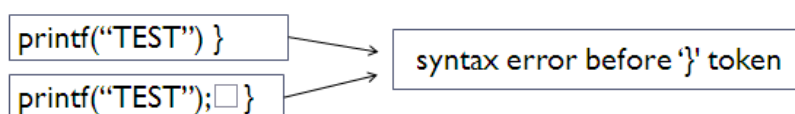


図3 エラーとコードの対応

4. 提案する手法

前章で述べた原因と、その理由に対して、支援を行う手法として、エラーメッセージに対応した訳文と解決策を提示するシステムを提案する。本システムは、ウェブ上で動作し、以下の3つの機能をブラウザを通じて、提供する。

4.1. 和訳エラーメッセージ検索機能

英文のエラーメッセージから日本語化されたエラーメッセージを検索、表示する。3.2の問題点を解決するため、この訳には意味を重視したものをを用いる。和訳はあらかじめデータベースに保存されており、入力されたエラーメッセージに対応したものが表示される。

4.2. 解決策提示機能

4.1で示した検索機能と連動し、エラーメッセージに対応した解決策を提示する。ここでの解決策とは、そのエラーメッセージが発生するサンプルソースコードと、なぜそのソースコードでエラーとなるのかを解説する文章を指す。これにより学習者の知識不足を補い、コンパイルエラーの修正を助ける。提示される解決策は指導者によってあらかじめ作られたデータベースから提供される。

4.3. 学習者からの情報収集機能

4.2で示した解決策提示機能には、提示できる解決策の数に限界があるという問題点が挙げられる。これは3.3で述べたように、エラーメッセージと修正方法が一对多の関係にあり、指導者が修正方法をあらかじめ網羅したデータベースを作成することが困難なためである。そこで学習者から以下の情報を投稿してもらうことで、より多くの解決策を提示可能にする。

- 発生したエラーメッセージ
- エラーが発生した状況を示すコード
- エラーへの解決策

これらの情報は、多くの解決策を提示するだけでなく、4.2の解決策を推敲することや、指導者が提供する解決策への採用など、今後のシステム改良にも役立つと考えられる。また、これらの投稿には、学習者が“参考になった”か“参考にならなかった”かの2値による評価を与えられるようになっており、同じエラーメッセージに複数の投稿がある場合は、この評価値の高いものから順に表示することで、学習者が、より効率的に解決策を見つけ出すことができる。

また、投稿の乱造や、評価の急変を防ぐため、投稿機能と評価機能についてはアカウント制とする。投稿内容の閲覧については制限を設けない。

4.4. システム動作のイメージ

学習者は、コンパイル時にエラーが出力されたとき、そのエラーメッセージを検索フォームに入力し、検索ボタンを押す。すると、システムはデータベースから該当するエラーメッセージに関する情報を入手し、図4のように利用者側へ提供する。図4の中段にある掲示板タグをクリックすることで図5のような掲示板の画面が表示される。ここでは4.3で述べた機能が提供される。

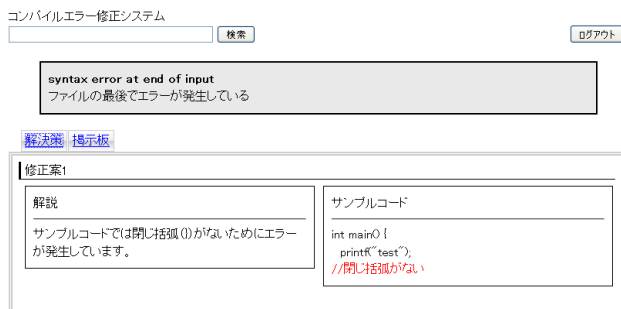


図4 システムのイメージ

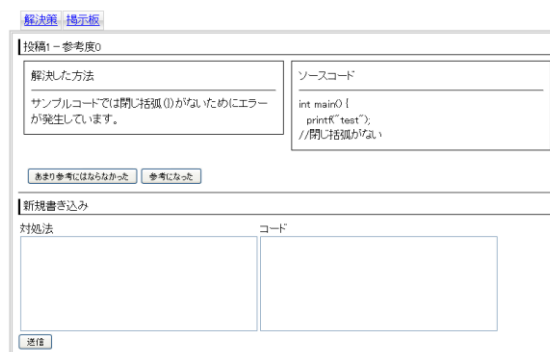


図5 掲示板イメージ

5. データベースと検索アルゴリズム

本システムの根幹となる2つの部分について以下で説明を行う。

5.1. エラーメッセージデータベース

本システムは予め指導者によって用意されたエラーメッセージと対処法のデータベースを用いる。このデータベースは、実際にC言語の演習授業において学生が発生させたエラーメッセージ約13000件を、C言語の統合開発環境AZUR[3]を用いることで取得し、それをもとに作成した。

5.2. エラーメッセージ検索アルゴリズム

本システムでのエラーメッセージの検索には正規表現でのマッチングを用いた。エラーメッセージには関数名など変化する箇所が含まれるため、入力されたエラーメッセージそのものによって検索することは難しい。そのため、5.1でのデータベースを作る際に、発生が予測されるエラーメッセージをあらかじめ正規化しておき、その正規化されたエラーメッセージと入力されたエラーメッセージを比較することによって検索を行っている。

6. 他のシステムとの相違点

6.1. Google 検索[1]

Googleでエラーメッセージを検索すると、他の質問サイトやハウツーサイトへのリンクが上位に来ているが、それらの中には間違った情報も含まれており、初学者にはその有用性の判断が難しい。一方、本システムは、コンパイルエラーを専門とし、指導者があらかじめ用意したデータベースによって稼働しているため、学習者にとって不利益となる情報が表示されることはない。

6.2. 山本らの研究[2]

あらかじめ定められた課題に対しての、エラー修正支援を目的としており、コンパイルエラーが発生した場合に、ソースコードに発生したエラーメッセージを全て取得し、その組み合わせによって原因の可能性と修正方法を示すことで学生を支援するが、これは既に決められた課題にしか適用できないという問題がある。これに対して、本システムは課題に依らない支援を行うため、指導者の負担を軽減できるほか、単一のエラーメッセージに対して4節で示したような情報を提示することで、修正方法だけでなく発生した原因まで理解できると考えられる。

7. 現状と今後

現状では、システムの検索から各情報を提供という流れまでと掲示板のシステムが完成している。現在は提供するデータとその提示方法について検討中である。また、本システムはAZURとの連動も視野に入れている。具体的には、AZURの利用者がコンパイルエラーが発生させた場合に、ブラウザを経由せず、AZURから本システムの情報が閲覧可能になることで、より容易に情報を閲覧可能にすることを目指す。

参考文献

[1] "Google", <http://www.google.co.jp/>

[2] 山本 耕大, 春原 将寿, 大金 克紀, 中村 勝一, 横山 節雄, 宮寺 庸造, "エラー要因事例ベースの動的学習手法を導入したC言語教育システムの開発と基礎的評価", 電子情報通信学会技術研究報告. ET, 教育工学 108(146), June 2008 pp.67-72

[3] Toshiyuki IMAIZUMI, Hiroaki HASHIURA, Saeko MATSUURA, and Seiichi KOMIYA, "A Programming Learning Environment "AZUR" : Visualizing Block Structures and Program Function Behavior", Proceedings of the 9th Joint Conference on Knowledge-Based Software Engineering (JCKBSE'10) August 2010 pp.306-311