

情報科学実験 B

担当：松本 真佑 先生

提出者：秋口 敬

コース：計算機科学コース

学籍番号：09B19002

提出年月日：2021 年 12 月 16 日

1 課題の目的

Pascal 風言語で記述されたプログラムが構文的に正しいかどうか判定すると共に、変数等の宣言や有効範囲の誤りや、演算子と被演算子の間や仮パラメータと実パラメータの間などでの型の不整合といった、意味的な誤りを発見するプログラムを作成する。

2 課題達成の方針と設計

2.1 方針

課題2で作成したプログラムに構文木を加え、その構文木を用いて意味エラーを一つ一つ検出していく。

2.2 設計

構文エラーを検出すると同時に構文木を作成する。
構文エラーがあれば出力する。
変数名や仮パラメータの内容を保存する。
重複した変数宣言があれば意味エラーを出力する。
宣言されていない変数が出てきたら意味エラーを出力する。
添え字の型が integer でなければ意味エラーを出力する。
代入文の左辺に配列型の変数名が指定されていたら意味エラーを出力する。
if 文や while 文の式の型が boolean でないなら意味エラーを出力する。
演算子と被演算子の間で型の不整合が発生していたら意味エラーを出力する。
代入文の左辺と右辺の式の間で型の不整合が発生しているなら意味エラーを出力する。
手続き名が、副プログラム宣言において宣言されていないなら意味エラーを出力する。

3 プログラムの実装方針

java のクラスを用いてクラス型の配列をリスト型にして構文木として機能させる。その後変数宣言や仮パラメータ宣言のある頂点をみて、変数の型や名前を保存する。その後、それぞれの意味エラーが起こりうる構文木の頂点から下の木をチェックしていく。

3.1 構文エラーを検出すると同時に構文木を作成する

構文が正しく作られていれば新たに頂点を作って番号を登録し、親と子のつながりを作る。この時、頂点のノードに定義 (仮パラメータ宣言や、変数名、定数など) やその字句、何番目のトークンかを登録する。これを構文解析とともに行う。

3.2 変数名や仮パラメータの内容を保存する、宣言されていない変数が出てきたら意味エラーを出力する

主プログラムおよび副プログラムごとに変数名を保存する。この時、現在見ている主プログラムあるいは副プログラム内に同じ名前の変数があればエラーを出力する。なければ、変数名、何番目のトークンで出てきたか、標準型か配列型か、その変数の型を保存する。

3.3 宣言されていない変数が出てきたら意味エラーを出力する

変数を参照しているときに先に保存した変数の情報を見ていき、そのリストになればエラーを出力する。

3.4 添え字の型が integer でなければ意味エラーを出力する

定義が添え字である頂点について、その添え字の下の変数および定数が boolean や char であるものが含まれたとき、エラーを出力する。

3.5 代入文の左辺に配列型の変数名が指定されていたら意味エラーを出力する

代入分の左辺が配列型変数であるものが来た時に、その子に添え字がなければエラーを出力する。

3.6 if 文や while 文の式の型が boolean でないなら意味エラーを出力する

条件文について、boolean 型の変数単体で来るか、あるいは関係演算子の入った式が来るはずなので、それ以外の場合の時、エラーを出力する。

3.7 演算子と被演算子の間で型の不整合が発生していたら意味エラーを出力する

定義が単純式の場合で、加法演算子がある場合、その項と項の型が同じかどうかを確認する。例えば変数なら、変数のリストと照らし合わせて型を確認し、定数なら数字か文字かなどを見て、それらを比較すればよい。

同じように、定義が項の場合で乗法演算子がある場合、その因子と因子が同じかどうか確認する。

3.8 代入文の左辺と右辺の式の間で型の不整合が発生しているなら意味エラーを出力する

左辺の変数が integer か char 型なら右辺の変数および定数が同じ型かどうかを確認する。左辺が boolean なら、右辺の式が関係演算子を持つ式かどうか確認する。そうでないなら右辺の変数および定数が boolean かどうか確認する。上記の条件を満たさなければエラーを出力する。

3.9 手続き名が、副プログラム宣言において宣言されていないなら意味エラーを出力する

副プログラム宣言時に、副プログラム名をリストに保存しておく。手続き呼び出しが行われている頂点についてその字句を保存している副プログラム名と比較する。もし一度も出てきてなければエラーを出力する。

4 まとめ

今回の課題について、構文木を作成することで条件文や代入分の箇所が楽ができた印象がする。変数や仮パラメータ名の箇所は場当たり的な実装になってしまった。また、木構造をうまくつけた実装を課題の後半から行ったため、前半の実装に時間がかかってしまった。

5 感想、謝辞

課題が進むにつれて難易度が徐々に高くなっており不安を感じる。Slack での質問に答えてくださった松本先生、TA の藤本さんに感謝の意を示します。