

プロジェクト演習

課題3

担当しているソースコード

nextch
printitem
simplestat
statements
strliteral

作業中ということは分かりましたが、説明が少ないので具体的な作業内容が分かりません・・・

どうしてそう書かれているのかを考えながら進めてください。while文と違って注意しなければならない点は、文列が“end”ではなく“until”で終わること、文の末尾に「式」があること、繰り返しは最低1回実行されること、などでしょうか。

repeatの実装の課題について

やったことに関する考察がきちんと書いてあれば良いレポートになるのですが・・・

statements.c内にrepeatStatementを作成した。whileStatementの中身を参考にしてどんな処理が行われているかわからない状態で複製したため、まだ変更を加える必要がある可能性が高い。シンボルの名前、ポインタの名前などは新しくしておいた。

statements.c内のfetchStatementでforStatementやwhileStatementに処理を写しているswitch文があったため、repeatStatementに処理を移す処理を追加した。

シンボルのnode_repeatとポインタ*repを宣言する必要がある。

シンボルやポインタの名前を変更しているが、while文の解析を行うプログラム文を複製しているので、コンパイルは通るはず。 **この説明だけでは、何のためにどんな作業をしたのか分かりません**

statements.cのcodeblock()内でcodeblock()が受け取ったトークンを正規のトークンだと認識できていなかったため、stat_setにsym_repeat, sym_untilを追加した。

repeat以外の拡張機能について

sym_until は追加する必要ないと思います。

配列の実装、組み込み関数の実装にあたり自分が担当しているソースコードで変更が必要だと考える。

strliteral.cで変数および関数のリテラルの宣言を解析しているであろうと考えているが、リテラルの宣言を実装するにはどのように変更を加えれば良いかがわかっていない。

strliteral.c は文字列リテラルしか扱っていないので、配列や組込関数の実装には関係しません。

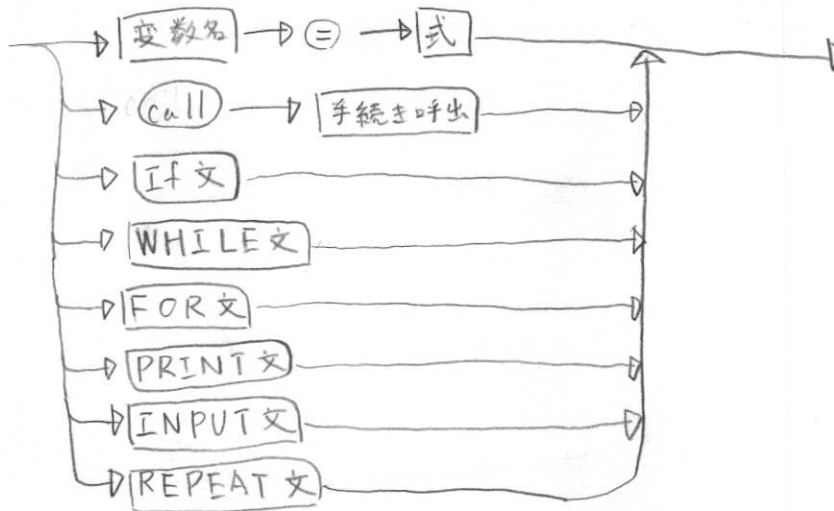
なお、思うように動作しなかったとき、ソースコードだけをぼーっと見るよりは、Xcodeでステップ実行を試みるなど、手を動かした方が意外と早く解決することもあります。さらに、複数人で手分けして原因を考えた方が早く解決するでしょう。

G07の報告書

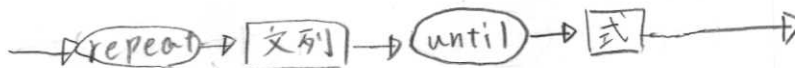
修正した構文図

文

実行文



REPEAT文



(1) repeat文の実装

予約語の追加をする

シンボル (トークン) の追加をする

repeatのnodeを作成

repeatのstatementを追加

execute内で実装する

実装の方針を説明して欲しいのです。

例えば「repeatのnode」とは何？

(4)動作しませんでした

動作しない、だけでは情報が不足。

コンパイルは通ったのか、実行時にエラーが出るのか。

実行はできるが想定と違う動作なのか。

G7