Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления» (ИУ)

КАФЕДРА «Теоретическая информатика и компьютерные технологии» (ИУ9)

ЛАБОРАТОРНАЯ РАБОТА №1

«Раскрутка самоприменимого компилятора»

Вариант 4

Выполнила:

студентка группы ИУ9-61Б

Бойко Маргарита Сергеевна

Москва, 2021

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ.....2
Цель работы..... 3
Исходные данные..... 3
Задание..... 3
Реализация.....4
Выводы..... 5

Цель работы

Целью данной работы является ознакомление с раскруткой самоприменимых компиляторов на примере модельного компилятора.

Исходные данные

В качестве модельного выберем компилятор BeRo Tiny Pascal, разработанный Бенжамином Рузо (Benjamin Rosseaux). Входным языком компилятора является язык Pascal, совместимый с диалектами Delphi 7 и FreePascal $\geq$ 3.0, а целевым языком — исполнимый код.

Исходный текст компилятора составлен на языке Pascal, совместимом с подмножеством диалектов Delphi 7 и FreePascal $\geq$ 3.0, при этом сам реализован на этом подмножестве. Тем самым, компилятор является самоприменимым.

Исходные данные для выполнения лабораторной работы в операционной системе Linux представлены следующим набором файлов:

btpc64.pas — исходный текст компилятора BeRo Tiny Pascal;

btpc64 — бинарная версия компилятора, полученная путём его раскрутки;

hello.pas — программа, предназначенная для проверки работоспособности компилятора.

Задание

Выполнение лабораторной работы заключается в осуществлении одного шага раскрутки самоприменимого компилятора BeRo Tiny Pascal и состоит из нескольких этапов:

1. Добавление во входной язык компилятора **btpc** новых возможностей (комментарии, начинающиеся с символа “?”) путём редактирования его исходного текста, в результате чего должен получиться файл **btpc64_2.pas**

(следует сначала скопировать **btpc64.pas** в **btpc64_2.pas**, а потом вносить в него правки).

2. Компиляция **btpc64_2.pas**, в результате которого должен получиться файл **btpc64_2**.
3. Проверка работоспособности **btpc64_2**, на небольшой программе, в которой обязательно должны использоваться новые возможности языка.
4. Внесение изменений в **btpc64_2.pas**, связанных с использованием новых возможностей языка, и сохранение новой версии исходного текста компилятора в файле **btpc64_3.pas**.
5. Завершение шага раскрутки путём компиляции **btpc64_3.pas** с помощью полученного на этапе 2 файла **btpc64_2**.
6. Разница между файлами **btpc64.pas** и **btpc64_2.pas** (отображаемая командой `diff -u btpc64.pas btpc64_2.pas`) должна демонстрировать изменения, внесённые в логику работы компилятора.
7. Разница между файлами **btpc64_2.pas** и **btpc64_3.pas** (отображаемая командой `diff -u btpc64_2.pas btpc64_3.pas`) должна демонстрировать новые возможности языка.

Реализация

На листинге 1 представлены различия **btpc64.pas** и **btpc64_2.pas**.

```
--- btpc64.pas  2020-02-15 14:28:10.000000000 +0300
+++ btpc64_2.pas        2021-02-13 14:37:20.350030279 +0300
@@ -795,6 +795,12 @@
  end else begin
   Error(102);
  end;
+ end else if CurrentChar='?' then begin
+  ReadChar;
+  repeat
```

```
+    ReadChar;
+   until (CurrentChar=#10) or (CurrentChar=#0);
+   GetSymbol;
  end else begin
   Error(102);
  end;
```

Листинг 1. Различия файлов btpc64.pas и btpc64_2.pas

На листинге 2 представлены различия **btpc64_2.pas** и **btpc64_3.pas**.

```
--- btpc64_2.pas        2021-02-13 14:37:20.350030279 +0300
     +++ btpc64_3.pas        2021-02-13 14:46:53.179823389 +0300
     @@ -575,6 +575,7 @@
       Halt;
      end;

     +? reading a character
      procedure ReadChar;
      begin
       if not EOF then begin
```

Листинг 2. Различия файлов btpc64_2.pas и btpc64_3.pas

Выводы

В результате выполнения лабораторной работы была произведена раскрутка компилятора BeRo Tiny Pascal. В него была добавлена возможность написания комментариев, начинающихся с “?”.

Была проведена проверка компилятора на непротиворечивость, поскольку он способен воспроизвести свой собственный код.