

Лабораторная работа № 1.1. Раскрытие самоприменимого компилятора

18 февраля 2025 г.

Артём Панкратов, ИУ9-61Б

Цель работы

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

Индивидуальный вариант

Компилятор Р5. Сделать так, чтобы символы в строке программы, расположенные справа от 100-й позиции, не учитывались (считались комментарием).

Реализация

Различие между файлами pcom.pas и pcom2.pas:

```
--- pcom.pas      2020-02-15 14:28:42.000000000 +0300
+++ pcom2.pas     2025-02-18 10:38:06.793030800 +0300
@@ -1335,6 +1335,14 @@
     test := eol;
     if test then nextch
 until not test;
+
+ if chcnt > 100 then
+ begin
+   repeat nextch until eol;
+   nextch;
+   goto 1;
+ end;
+
if chartp[ch] = illegal then
  begin sy := othersy; op := noop;
    error(399); nextch
```

Различие между файлами pcom2.pas и pcom3.pas:

```
--- pcom2.pas    2025-02-18 10:38:06.793030800 +0300
+++ pcom3.pas    2025-02-18 10:41:53.112636700 +0300
@@ -1336,7 +1336,7 @@
     if test then nextch
     until not test;

-   if chcnt > 100 then
+   if chcnt > 100 then
       begin
         repeat nextch until eol;
         nextch;
@@ -1759,7 +1759,8 @@
         scalar: begin write(output, 'scalar':10);
                   if scalkind = standard then
                     write(output, 'standard':10)
-                   else write(output, 'declared':10, ' ':4, ctptoint(*ord*)(fconst):int
+                   else
+                     write(output, 'declared':10, ' ':4, ctptoint(*ord*)(fconst):intsize
                     writeln(output)
                   end;
         subrange: begin
@@ -1778,23 +1779,25 @@
                   followstp(elset)
                 end;
         arrays: begin
-           writeln(output, 'array':10, ' ':4, stptoint(*ord*)(aeltype):intsize(
-             stptoint(*ord*)(inxtype):6);
+           writeln(output, 'array':10, ' ':4,
+             stptoint(*ord*)(aeltype):intsize(*6), ' ':4, stptoint(*ord*)(inx
             followstp(aeltype); followstp(inxtype)
           end;
         records: begin
-           writeln(output, 'record':10, ' ':4, ctptoint(*ord*)(fstfld):intsize(
+           writeln(output, 'record':10, ' ':4, ctptoint(*ord*)(fstfld):intsize(*6
             stptoint(*ord*)(recvar):intsize(*6)); followctp(fstfld);
             followstp(recvar)
           end;
         files: begin write(output, 'file':10, ' ':4, stptoint(*ord*)(filtype):intsize
                   followstp(filtype)
                 end;
-         tagfld: begin writeln(output, 'tagfld':10, ' ':4, ctptoint(*ord*)(tagfieldp):i
+         tagfld: begin
+           writeln(output, 'tagfld':10, ' ':4, ctptoint(*ord*)(tagfieldp):intsi
             ' ':4, stptoint(*ord*)(fstvar):intsize(*6));
```

```

        followstp(fstvar)
    end;
-   variant: begin writeln(output, 'variant':10, ' ':4, stptoint(*ord*)(nxtvar):int
+   variant: begin
+       writeln(output, 'variant':10, ' ':4, stptoint(*ord*)(nxtvar):intsize
        ' ':4, stptoint(*ord*)(subvar):intsize(*6*), varval.ival);
        followstp(nxtvar); followstp(subvar)
    end
@@ -1835,7 +1838,8 @@
        else write(output, 'formal':10);
        write(output, ' ':4, ctptoint(*ord*)(next):intsize(*6*), vlev, ' ':4, vad
    end;
-   field: write(output, 'field':10, ' ':4, ctptoint(*ord*)(next):intsize(*6*), ' ':4
+   field: write(output, 'field':10, ' ':4,
+       ctptoint(*ord*)(next):intsize(*6*), ' ':4, fldaddr:6);
    proc,
    func: begin
        if klass = proc then write(output, 'procedure':10)

```

Тестирование

Тестовый пример:

```

program test(output);
var
    a: integer;
begin
    a := 10;
    if a > 5 then writeln('a > 5')
    else writeln('a <= 5');
end.

```

Вывод тестового примера на stdout

```
a > 5
```

Вывод

При выполнении лабораторной работы мы изучили устройство компилятора P5 и научились при помощи раскрутки создавать самоприменимый компилятор.