# PL语言编译程序实验项目测试用例说明

## 测试用例For

* 测试用例文件

for.pls

* 测试目的

对for中to、downto、表达式做循环条件以及两重循环进行测试

* 程序说明

program main;  
 var d,a,b,k,j,i:integer;  
 data : record  
 i : integer;  
 j : integer;  
 k : integer;  
 s: real;  
 name: char;  
 end;  
 function Fibonacci(m, n : integer) : integer;  
 begin  
 if m>100  
 then Fibonacci := m  
 else  
 Fibonacci := Fibonacci(m+n,m);  
 end;  
 function isOdd(m : integer) : integer;  
 begin  
 if (m mod 2)=1  
 then isOdd := 1  
 else  
 isOdd := 0;  
 end;  
begin  
 data.i := 1;  
 data.j := 1;  
 data.k := 9;  
 data.s:=36.6636;  
 data.name:='S';  
  
 d:=Fibonacci(data.i,data.j);  
 call write(d);  
  
 j:=1;  
 i:=1;  
 repeat  
 k := j ;  
 j := i+j;  
 i := k  
 until j >100;  
 call write(j);  
  
 for a:= 0 to data.k do  
 call write(isOdd(a));  
  
 case data.name of  
 'S' : call write (11);  
 'M' : call write (00)  
 end  
  
  
end.

* 预期输出结果：

START PL/0

k

o

10

9

8

7

END PL/0

## 测试用例Repeat

* 测试用例文件

repeat.pls

* 测试目的

对repeat进行测试

* 程序说明

输出10-100之间的斐波那契数

program main;  
 var k,i,j:integer;  
begin  
 j := 1;  
 i := 1;  
 repeat  
 if ((j < 100) and (j>10))  
 then call write(j);  
 k := j ;  
 j := i+j;  
 i := k  
 until j >100  
end.

* 预期输出结果：

START PL/0

13

21

34

55

89

END PL/0

## 测试用例Case

* 测试用例文件

case.pls

* 测试目的

对分支条件表达式、多条件进行测试

* 程序说明

做两次case来进行分支语句的判断

program main;  
 var d,a:integer;  
begin  
 a := 2;  
 case a\*2 of  
 2 : d:=0;  
 4 : d:=1;  
 6 : d:=2  
 end;  
 case d of  
 1,4 : call write (-1);  
 2,5 : call write (0);  
 3,6,7 : call write (1)  
 end  
end.

* 预期输出结果：

START PL/0

-1

END PL/0

## 测试用例Func

* 测试用例文件

func.pls

* 测试目的

对函数声明、定义、前后序相互调用、递归调用功能进行测试

* 程序说明

做斐波那契求出大于10的第一个数（13），然后求此数二倍（26）与64的最大公约数

program main;  
 var c,a,b:integer;  
  
 function GCD1 (m, n : integer) : integer ; forward;  
 function GCD2 (m, n : integer) : integer ; forward;  
  
 function GCD1 (m, n : integer): integer;  
 begin  
 if n=0 then GCD1 := m else GCD1 := GCD2(n, m mod n);  
 end;  
  
 function GCD2 (m, n : integer): integer;  
 begin  
 if n=0 then GCD2 := m else GCD2 := GCD1(n, m mod n);  
 end;  
  
 function Fibonacci(m, n : integer) : integer;  
 begin  
 if m>10  
 then Fibonacci := m  
 else  
 Fibonacci := Fibonacci(m+n,m);  
 end;  
  
begin  
 a := 1;  
 b := 1;  
 c := Fibonacci(a,b);  
 call write(c);  
 a := GCD2(c\*2,64);  
 call write(a)  
end.

* 预期输出结果：

START PL/0

13

2

END PL/0

## 测试用例Real

* 测试用例文件

real.pls

* 测试目的

对实型的表达式运算、判断、作为参数在函数中传入传出功能进行测试

* 程序说明

做实型检查，分别是直接实型和科学计数法的测试，若d与0.05的差距在0.0001以内则输出1

program main;  
 var d,a:real;  
 function foo (m: real ) : real;  
 begin  
 m := m -0.00001;  
 foo := m + 3e-2  
 end;  
begin  
 a := 2e-2;  
 d := foo(a);  
 call write(d);  
 if ((d > 0.05) and (d - 0.05 < 0.0001)) or ((d < 0.05) and (0.05 - d < 0.0001)) then  
 call write(1)  
 else  
 call write(0)  
end.

* 预期输出结果：

START PL/0

0.04999

1

END PL/0

## 测试用例Recd

* 测试用例文件

recd.pls

* 测试目的

对记录类型的表达式运算、判断、分支功能进行测试

* 程序说明

对记录类型进行赋值与判断，最后输出记录类型中casetest22的值

program main;  
 var shape : integer;  
 var date : record  
 year : real;  
 month : real;  
 day : real;  
 case shape: integer of  
 1: (casetest1, casetest2: integer);  
 2: (casetest11, casetest22: real);  
 end;  
  
begin  
 shape := 2;  
  
 date.day := 365;  
 date.month := date.day/30;  
 date.year := date.month / 12;  
  
  
 if ((date.year > 1.0138) and (date.year - 1.0138 < 0.0001)) or ((date.year < 1.0138) and (1.0138 - date.year < 0.0001)) then  
 if (shape = 1) then  
 begin  
 date.casetest1 := 1;  
 date.casetest2 := -1;  
 call write(date.casetest2)  
 end  
 else  
 begin  
 date.casetest11 := 0.01;  
 date.casetest22 := -1.01;  
 if (date.casetest11>0.0) then  
 call write(date.casetest22)  
 end  
  
 else  
 call write(0)  
  
end.

* 预期输出结果：

START PL/0

-1.01

END PL/0

## 测试用例Part1

* 测试用例文件

part1.pls

* 测试目的

对case嵌套for和repeat进行测试

* 程序说明

前面加和为10，然后倒序输出其二进制1010

program main;  
 var i,j,index,temp,k,x:integer;  
begin  
 k:=0;  
 x:=10;  
 for index:= 1 to x do  
 begin  
 case index mod 3 of  
 2: k:=k+0;

1: k:=k+1;  
 0: k:=k+2  
 end  
 end;  
 call write(k);  
  
  
 repeat  
 case k mod 2 of  
 1: call write (1);  
 0: call write (0)  
 end;  
 k:=k/2  
 until k <2;  
 call write (k);  
  
end.

* 预期输出结果：

START PL/0

10

0

1

0

1

END PL/0

## 测试用例Types

* 测试用例文件

types.pls

* 测试目的

对各个类型的表达式运算、判断、分支功能进行测试

* 程序说明

定义两个包含实型、整形、字符型的记录类型的每一项进行分别判断最后输出结果（字符型按ASCII码值算排序）

program main;  
 var date1 : record  
 year : integer;  
 month : integer;  
 day : integer;  
 s: real;  
 name: char;  
 end;  
  
 var date2 : record  
 year : integer;  
 month : integer;  
 day : integer;  
 s: real;  
 name: char;  
 end;  
  
begin  
 date1.day := 14;  
 date1.month := 2;  
 date1.year := 2000;  
 date1.s:=0.00019;  
 date1.name:='K';  
  
 date2.day := 15;  
 date2.month := 3;  
 date2.year := 1998;  
 date2.s:=2e-4;  
 date2.name:='O';  
  
  
 if (date1.year > date2.year)then  
 call write(1)  
 else  
 call write(0);  
  
 if (date1.month > date2.month) then  
 call write(1)  
 else  
 call write(0);  
  
 if (date1.day > date2.day)then  
 call write(1)  
 else  
 call write(0);  
  
 if (date1.s > date2.s)then  
 call write(1)  
 else  
 call write(0);  
 if(date1.name=date2.name)then  
 call write(1)  
 else  
 call write(0)  
end.

* 预期输出结果：

START PL/0

1

0

0

0

0

END PL/0

## 测试用例All

* 测试用例文件

all.pls

* 测试目的

对所有新增结构进行测试

* 程序说明

做两次斐波那契求出大于10的第一个数，第一次用记录类型的变量通过递归调用函数来进行，第二次用repeat来验算；之后for循环遍历调用函数对循环变量进行case语句检查；最后进行实型参数函数的检查；

program main;  
 var d,a,b,k,j,i:integer;  
 data : record  
 i : integer;  
 j : integer;  
 k : integer;  
 s: real;  
 name: char;  
 end;  
 function Fibonacci(m, n : integer) : integer;  
 begin  
 if m>10  
 then Fibonacci := m  
 else  
 Fibonacci := Fibonacci(m+n,m);  
 end;  
 function encode(m : integer) : integer;  
 begin  
 case m mod 4 of  
 0 : call write (0);  
 1 : call write (1);  
 2 : call write (10);  
 3 : call write (11)  
 end  
 end;  
  
 function fun(flag, n,m: integer) : integer;  
 begin  
 if flag=0  
 then fun := encode(n)  
 else  
 fun := Fibonacci(n,m)  
 end;  
  
 function print(n: real) : integer;  
 begin  
 call write(n);  
 print := 1;  
 end;  
begin  
 data.i := 1;  
 data.j := 1;  
 data.k := 3;  
 data.s:=3e-4;  
 data.name:='S';  
  
 d:=fun(1,data.i,data.j);  
 call write(d);  
  
 j:=1;  
 i:=1;  
 repeat  
 k := j ;  
 j := i+j;  
 i := k  
 until j >10;  
 call write(j);  
  
 for a:= 0 to data.k do  
 i:=fun(0,a,0);  
 i:=print(data.s-1.01);  
 case data.name of  
 'S' : call write (11);  
 'M' : call write (00)  
 end  
  
  
end.

* 预期输出结果：

START PL/0

13

13

0

1

10

11

-1.0097

11

END PL/0