



Интегрированная среда программирования Free Pascal

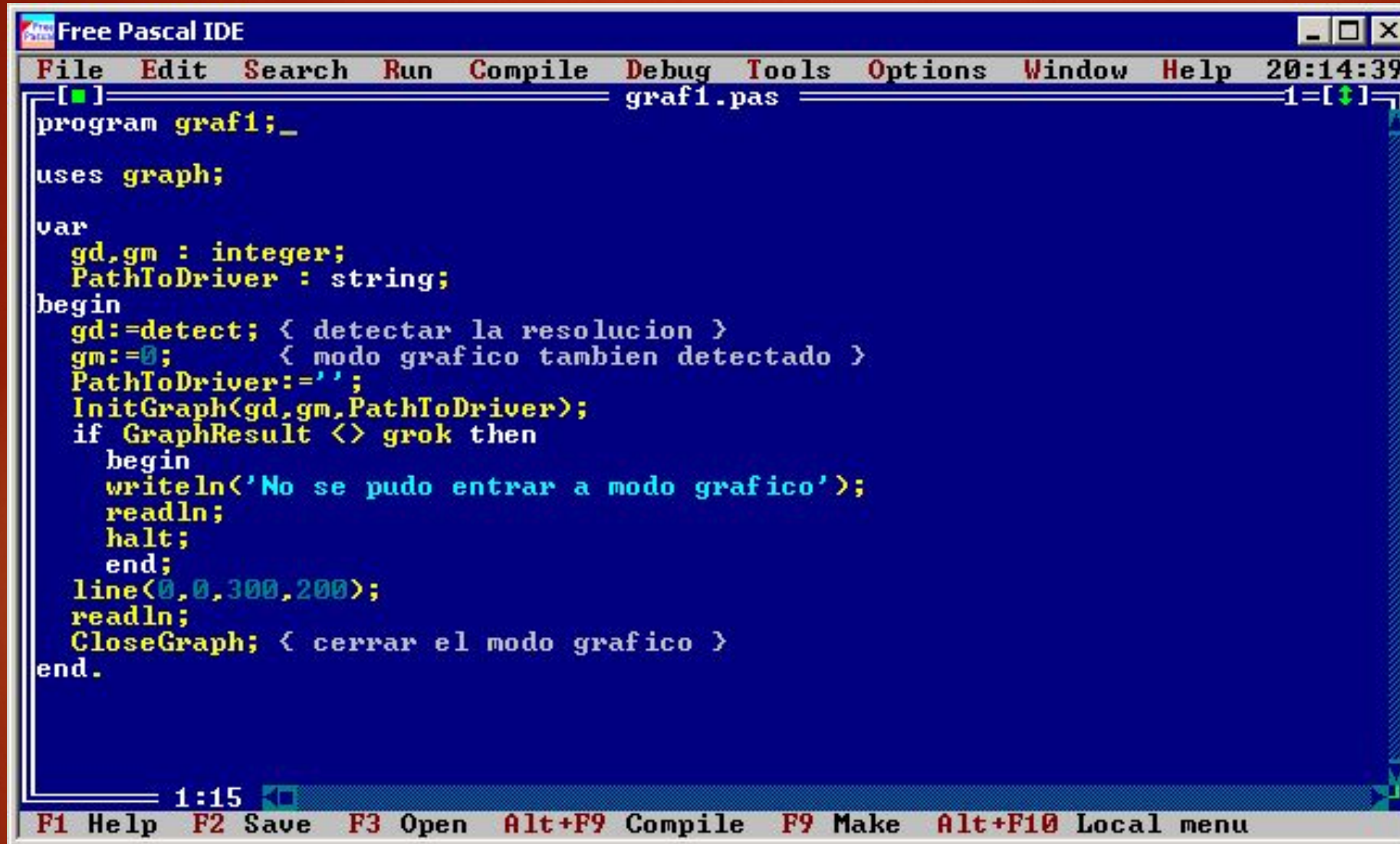
АВТОР – ВЛАДИСЛАВ КУРУЦ

8-Г КЛАСС

2. Содержание:

- I. Среда программирования
- II. Синтаксис языка
- III. Операторы
- IV. Ввод/Вывод данных
- V. Типы данных
- VI. Величины
- VII. История создания
- VIII. Структура программы
- IX. Примеры

3. Среды программирования:

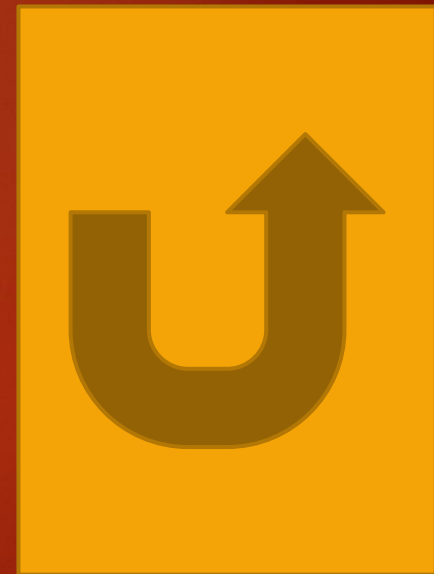


```
Free Pascal IDE
File Edit Search Run Compile Debug Tools Options Window Help 20:14:39
[ ] graf1.pas 1=[ ]
program graf1;_
uses graph;
var
  gd,gm : integer;
  PathToDriver : string;
begin
  gd:=detect; < detectar la resolucio n >
  gm:=0; < modo grafico tambien detectado >
  PathToDriver:='';
  InitGraph(gd,gm,PathToDriver);
  if GraphResult <> grok then
  begin
    writeln('No se pudo entrar a modo grafico');
    readln;
    halt;
  end;
  line(0,0,300,200);
  readln;
  CloseGraph; < cerrar el modo grafico >
end.
1:15
F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu
```



4. Синтаксис языка:

- I. Латинский алфавит
- II. Цифры
- III. Специальные значения
- IV. Служебные слова

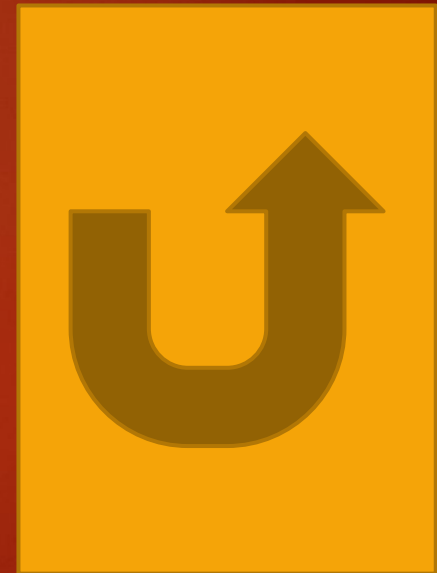


5. Операторы:

Pascal состоит из выполняющихся последовательно отдельных стандартных операторов.

В операторах `while`, `for`, `if`, `case` в качестве выполняемого оператора может использоваться *блок*. Такая конструкция, представляющая собой обычный оператор или блок, называется *сложным оператором*.

В `turbo pascal` для управления процессом компиляции существуют директивы, которые помещаются в комментарии и позволяют переключать режимы работы компилятора — например, включать и отключать проверку операций ввода-вывода, переполнения.



6. Ввод/Вывод данных:

Writeln (список вывода) ; — стандартная процедура, которая используется для вывода данных на экран. Значения выводятся в порядке их записи в списке вывода. Затем курсор переходит в начало следующей строки.

Write (список вывода) ; — стандартная процедура для вывода данных на экран без перевода курсора на новую строку.

Ввод данных в языке программирования паскаль обеспечивается процедурами `read()` и `readln()`. Ввод данных осуществляется либо с клавиатуры, либо из файла. Здесь рассматривается только ввод с клавиатуры.



7. Типы данных

В стандартном и расширенном паскале есть такие простые типы: числа с плавающей запятой(real), целые(integer), символьные(char), логические(boolean), и перечисления (конструктор нового типа введенный в pascal);



8. Величины:

ВИДЫ ВЕЛИЧИН

ПЕРЕМЕННЫЕ

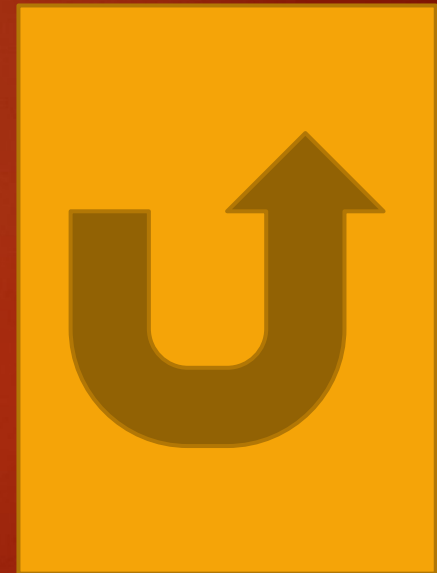
КОНСТАНТЫ



9. История создания:

Язык паскаль был создан Никлаусом Виртом в 1968—1969 годах после его участия в работе комитета разработки стандарта языка алгол-68. Язык назван в честь французского математика, физика, литератора и философа Блеза Паскаля, который создал первую в мире механическую машину, складывающую два числа. Первая публикация Вирта о языке датирована 1970 годом, представляя язык, автор указывал в качестве цели его создания — построение небольшого и эффективного языка, способствующего хорошему стилю программирования, использующему структурное программирование и структурированные данные.

Последующая работа Вирта была направлена на создание на основе паскаля языка системного программирования, с сохранением возможности вести на его базе систематический, целостный курс обучения профессиональному программированию. Результат этой работы — язык модуля- 2.



10. Структура программы:

```
Program ...;
```

```
    var ...;
```

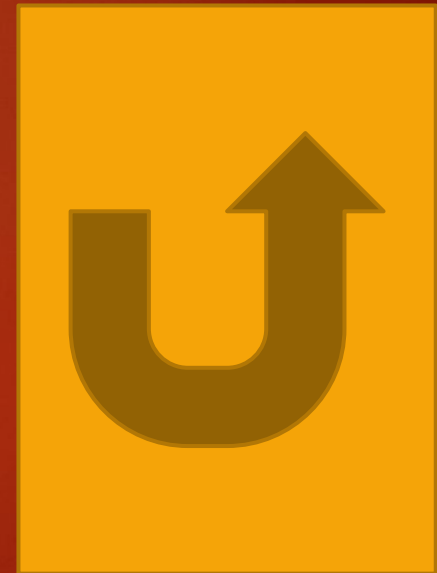
```
Begin;
```

```
    Readln(...);
```

```
    Writeln(...);
```

```
    Readln(...);
```

```
End;
```



11. Пример:

Условие: Найти длину окружности L и площадь круга S заданного радиуса R :
 $L = 2 \cdot \pi \cdot r$, $S = \pi \cdot r^2$.
В качестве значения π использовать 3.14.

Решение:

```
Program kurchix;  
const  
  pi = 3.14; { <-- Число "пи" }  
  
var  
  R, L, S: real;  
  
begin  
  write('R = ');  
  readln(R); { <-- Вводим радиус }  
  L := 2 * pi * R; { <-- Вычисляем длину окружности }  
  S := pi * sqr(R); { <-- Вычисляем площадь круга }  
  writeln('Результат:');  
  writeln(' L = ', L:0:2);  
  writeln(' S = ', S:0:2);  
  readln  
end.
```

