

# Структура программы на Паскале

<b>Program</b> [    ] ;	(заголовок программы)
<b>Var</b> [    ] ;	(раздел описания переменных)
<b>Begin</b>	(начало)
<b>Раздел операторов ;</b>	(команды)
<b>End.</b>	(конец)

# Операторы ввода и вывода информации



**Read** - (читать)

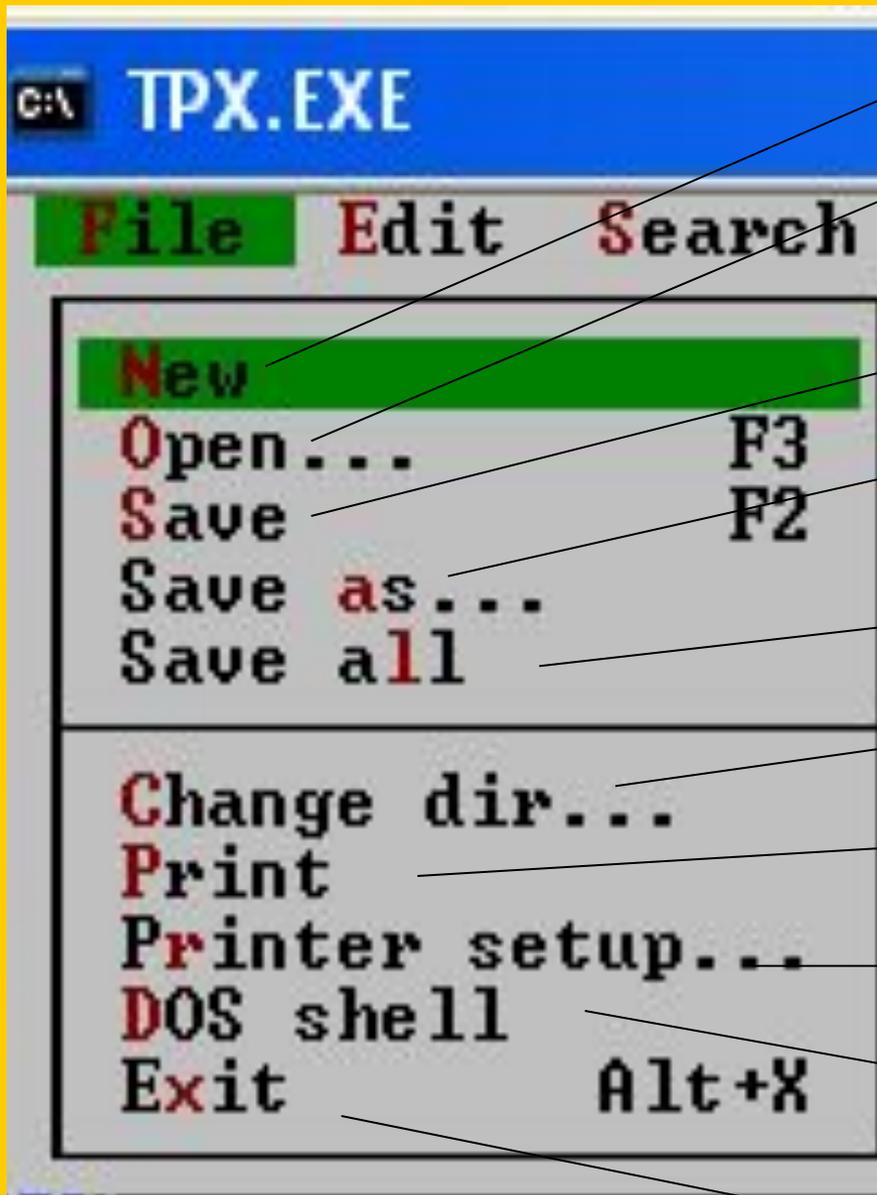
**Readln** - (читать строку)

**Write** - (писать)

**Writeln** - (писать в строку)

## Пример программы

<b>Program test;</b>	- имя программы
<b>Var A,B,C: integer;</b>	- список переменных: тип
<b>Begin</b>	- начало раздела операторов
<b>Readln (A,B);</b>	- оператор ввода переменных
<b>C:=A+B;</b>	- оператор присваивания
<b>Writeln (C);</b>	- оператор вывода
<b>Readln</b>	- чтение ответа
<b>End.</b>	- окончание программы



Создать новый файл

Открыть существующий файл

Сохранить файл (сохраняет файл активного окна)

Сохранить файл под именем

Сохраняет все измененные файлы

Изменить текущий каталог

Распечатать файл

Установить параметры принтера

Выход в операционную систему

Выход (прекращает работу в интегрированной среде)

# Оформление текста на экране

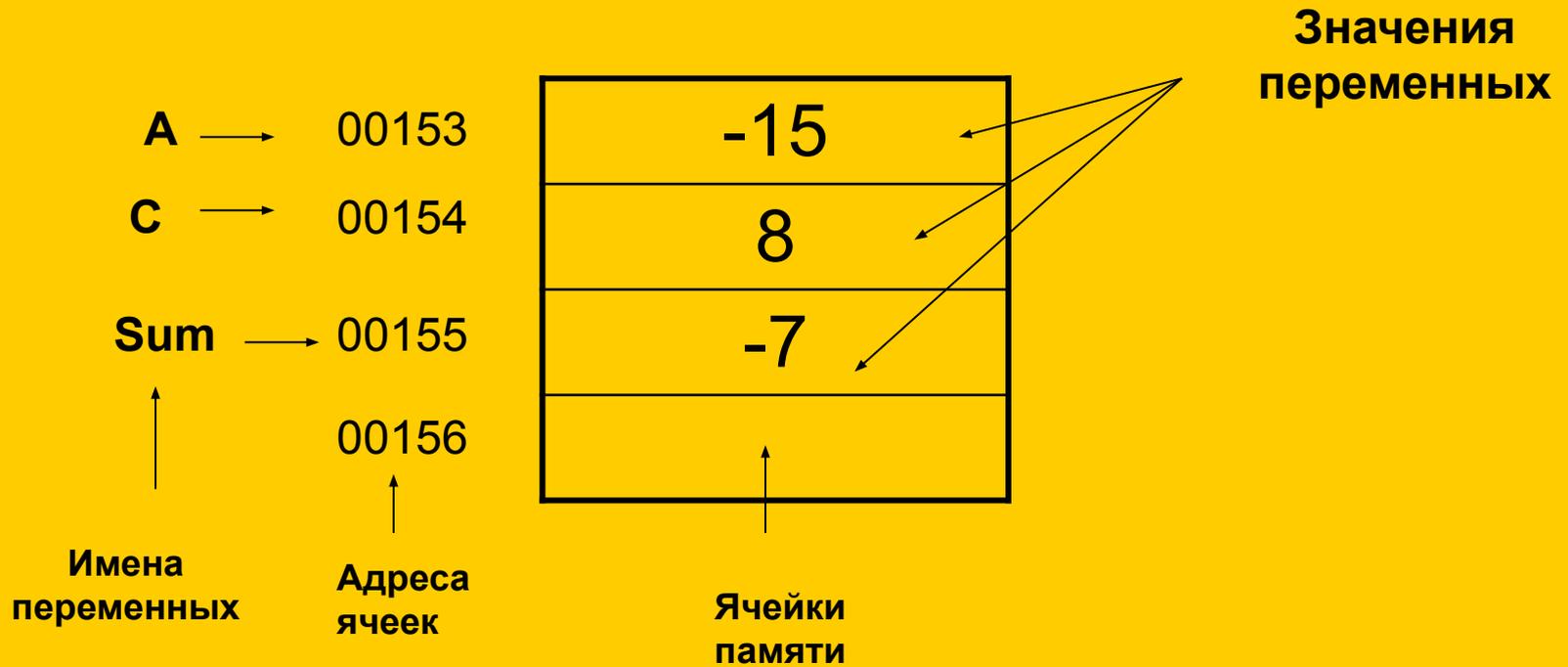
**Модуль CRT**- (это английская аббревиатура, обозначающая электронно-лучевую трубку – название модуля подчеркивает, что он умеет управлять способами вывода на экран).

**Модуль CRT**- не входит в стандарт языка, он является расширением возможностей Паскаля.

**Модуль CRT** содержит набор программ (процедур), которые позволяют:

- ✓ Задавать цвет символов;
- ✓ Очищать экран;
- ✓ Устанавливать курсор в любую позицию экрана;
- ✓ Выполнять множество других полезных действий.

# Хранение данных в памяти компьютера



Основная память состоит из ячеек (байтов), каждая из которых имеет адрес, то есть порядковый номер. В этих ячейках и хранятся данные.

Стандартная длина ячейки равна 8 бит.

# Представление целых чисел в памяти компьютера

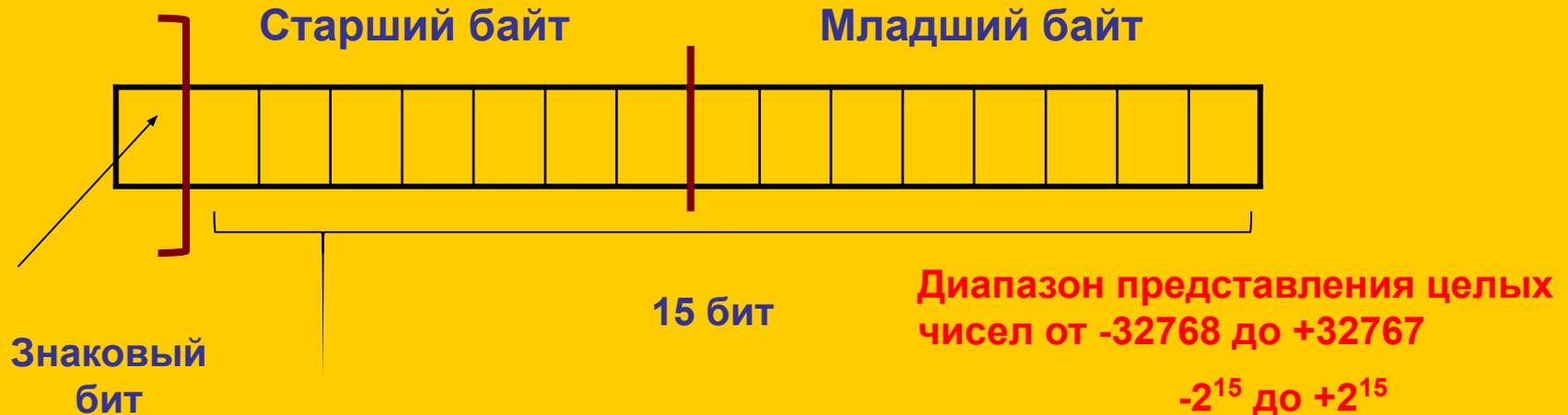
Для переменной типа `integer` выделяется ячейка длиной в 2 байта, что равно 16 битам. Такая ячейка получает символическое имя – имя переменной.

Крайняя левая позиция выделяется для знака числа:

0 – число положительное

1 – число отрицательное.

Остальные 15 позиций выделяются для записи самого числа в двоичном виде.



# Представление переменных вещественного типа

## Форматы записи вещественных переменных

0,7 = 0.7 или .7

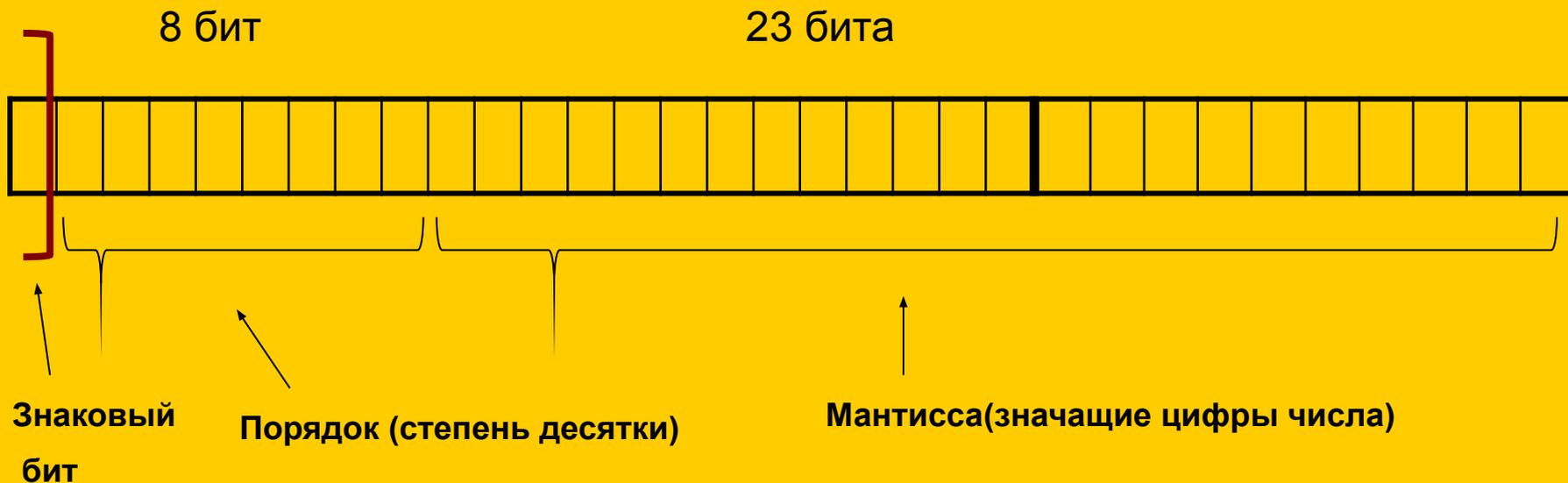
-2,1 = -2.1

2700=2,7\*10<sup>3</sup>

число **10** записывается в виде буквы E, а за ней идет величина степени **2.7E3**

**0,002=2\*10<sup>-3</sup>** соответствует запись

**2E-3**



# Описание вещественного типа данных (Real)

```
Program z1;
```

```
Var
```

```
A,b,c:real;
```

```
A:=3.5;
```

```
B:=7.6;
```

```
C:=A+B;
```

```
Writeln('сумма='C);
```

```
Readln;
```

```
End.
```

# Принцип работы с модулем CRT

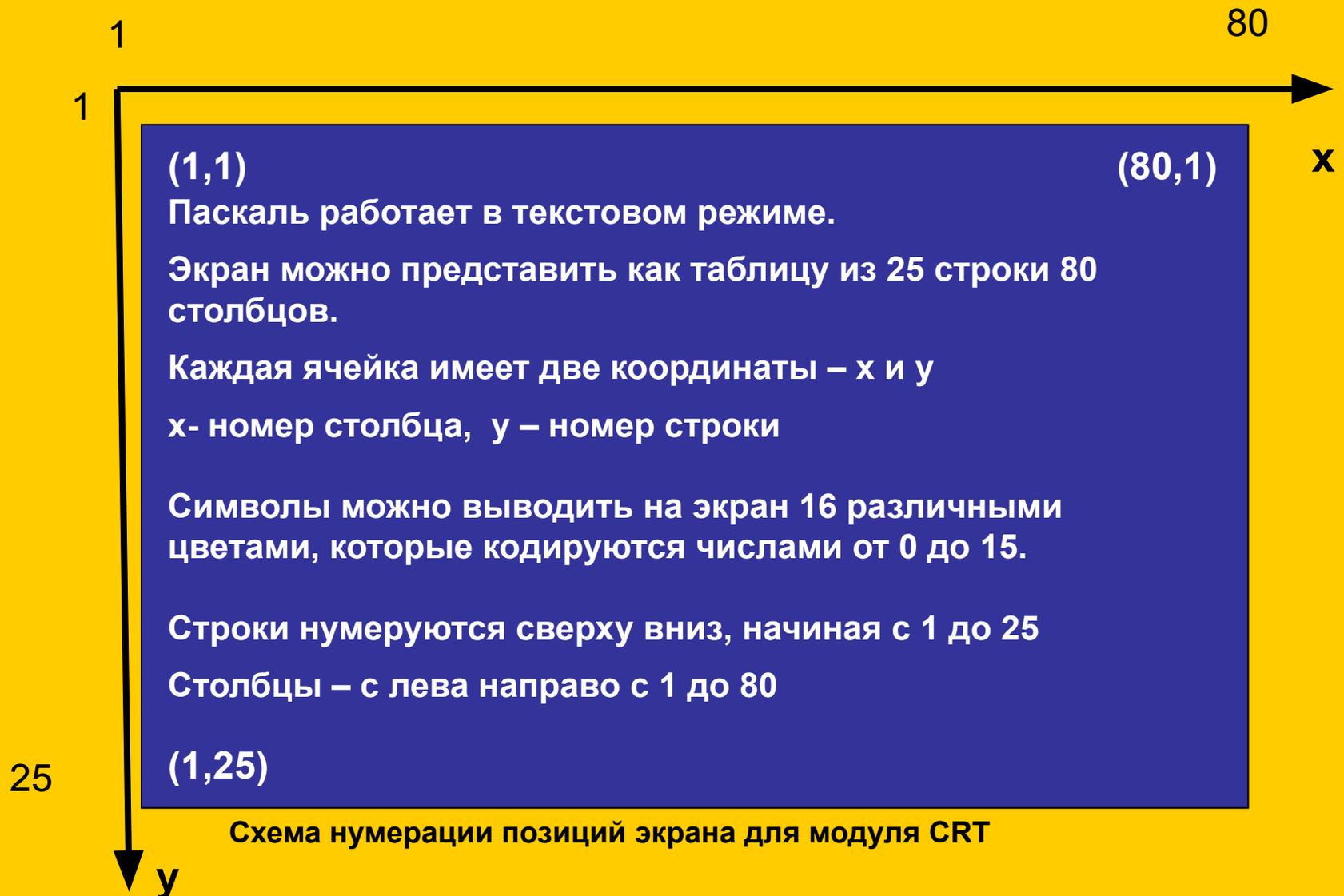


Схема нумерации позиций экрана для модуля CRT

# Использование модуля CRT

Для использования дополнительных библиотечных функций, необходимо в начале программы указать это в специальной секции объявления библиотечных модулей.

Она начинается словом uses

Затем через запятую перечисляются подключаемые модули.

## ФУНКЦИИ

**TextBackground(3);** – вызов процедуры для фонового цвета

**ClrScr ;** - процедура очистки экрана

**TextColor(14);** - процедура выбора цвета

**GoToXY (40,10)** - процедура установки курсора в точку экрана

**Deley (1000)** - процедура временной задержки на 1000 мкс.

# Пример программы

```
File Edit Search Run Compil
[ ]
program text;
uses crt;
begin
textbackground (3);
clrscr;
textcolor (14);
gotoxy (30,10);
writeln('все отлично');
readln
end.
```

C:\ TPX.EXE

File Edit Search Run Compile Debug Tools Options Window Help

```
NONAME00.PAS 1=[↑]
program bit;
var s,sk:real;
begin

writeln('введите стоимость покупки');
  readln(s);
if s>1000 then begin sk:=s-s*0.1; writeln('вам предоставляется скидка 10%');
writeln('стоимость покупки с учетом скидки',sk);end
  else begin sk:=s; writeln ('стоимость покупки без скидки',sk);end;
readln
end.
```

\* 11:1

F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu