

**Уважаемые студенты групп!**

**Вашему вниманию представлена лабораторная работа на тему  
«СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ». Работа рассчитана на 2 часа**

### **Задание**

1. Реализовать в системе Паскаль приведенный пример выполнения задания, протестировать программу и исправить ошибки.
2. Лабораторные работы оформляются в тетради в клеточку!
3. Дата предоставления фотоотчет до 31.03.2023
4. С уважением Ганзенко Ирина Владимировна

!!! Если возникнут вопросы обращаться по телефону 0721134803 (вацап),  
+79591134803 (телеграмм)  
[disobuch.ganzenko2020@mail.ru](mailto:disobuch.ganzenko2020@mail.ru)

## **СИМВОЛЬНЫЕ ПЕРЕМЕННЫЕ**

**Цель работы:** Изучение программ с использованием символьных переменных.

### **1 Теоретическое положение**

Для представления отдельных символов (например, литер, цифровых символов и др.) Язык Pascal имеет стандартный тип `char` (от англ. Character - символ, знак). Кроме числовой информации компьютер может обрабатывать, символьную информацию. Pascal оперирует с символьной информацией, которая может быть представлена отдельными символами или строками символов.

Для хранения и обработки отдельных символов используются переменные типа `char`. Значением переменной типа `char` может быть любой символ.

Переменная символьного типа должна быть объявлена в разделе описания переменных следующим образом:

Имя: `char`;

где имя - имя переменной символьного типа; `char` - ключевое слово - признак символьного типа.

примеры:

`a: char`;

`Kot: char`;

Переменным типа **CHAR** можно присваивать любые символы, вводимые с клавиатуры и отображать на экране монитора. Каждый такой символ имеет свой внутренний код (номер). При этом буквы расположены в алфавитном порядке, цифры - в порядке роста. Поэтому любой символ имеет

предшествующий ему символ (который имеет номер, на единицу меньше) и следующий (номер, на единицу больше).

С помощью 1 байта можно закодировать 28 символов = 256 символов.

Коды от

0 - 127 - универсальные (шаблонные символы и латинские буквы)

128 - 255 - русские буквы (большие и малые), или другая азбука.

**Символы (Char)** - это стандартный тип данных в дополнение к Integer, Real, String.

**Переменная типа Char** - это один символ.

Переменную можно превратить в код и наоборот с помощью специальных функций.

**Функция CHR (код)** - превращает число в символ (код -> символ).

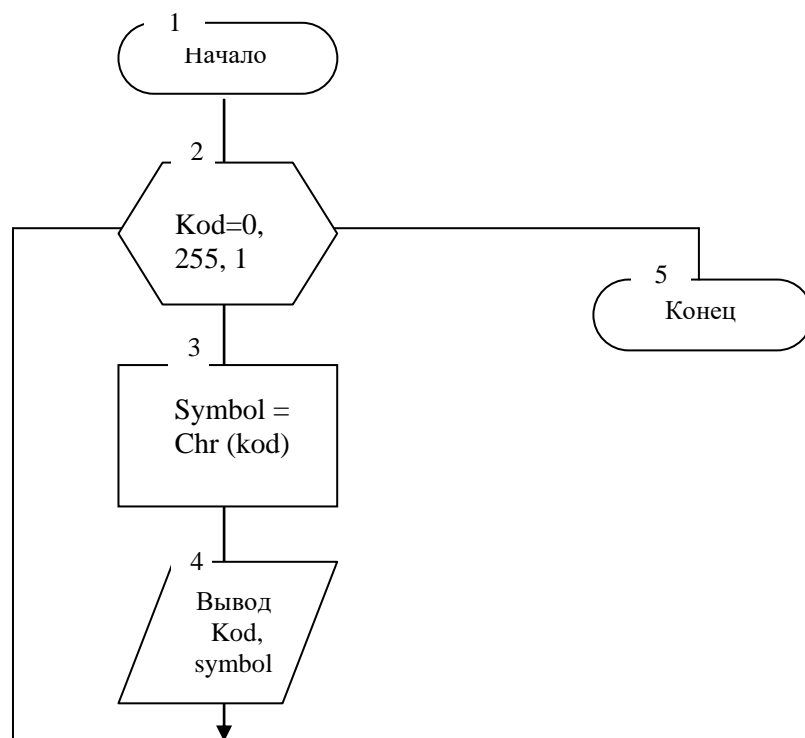
Если вместо кода взять число от 0 до 255 то по этой функции получим соответствующий символ.

**Функция Ord (символ)** - превращает символ в код (символ -> код).

## 2 Пример программы для вывода на экран всех символов с кодами от 0 до 255

Далее приведена блок-схема алгоритма и программа вывода на экран всех символов с кодами от 0 до 255.

### 2.1 Блок - схема алгоритма



### 2.2 Текст программы

```

program simbol;
uses crt;
var
kod:byte; simbol: char;
begin
    clrscr;
    for kod:=0 to 255 do
    begin
        simbol:=chr(kod);
        writeln(kod,'=', simbol);
    end;
    readln;
end.

```

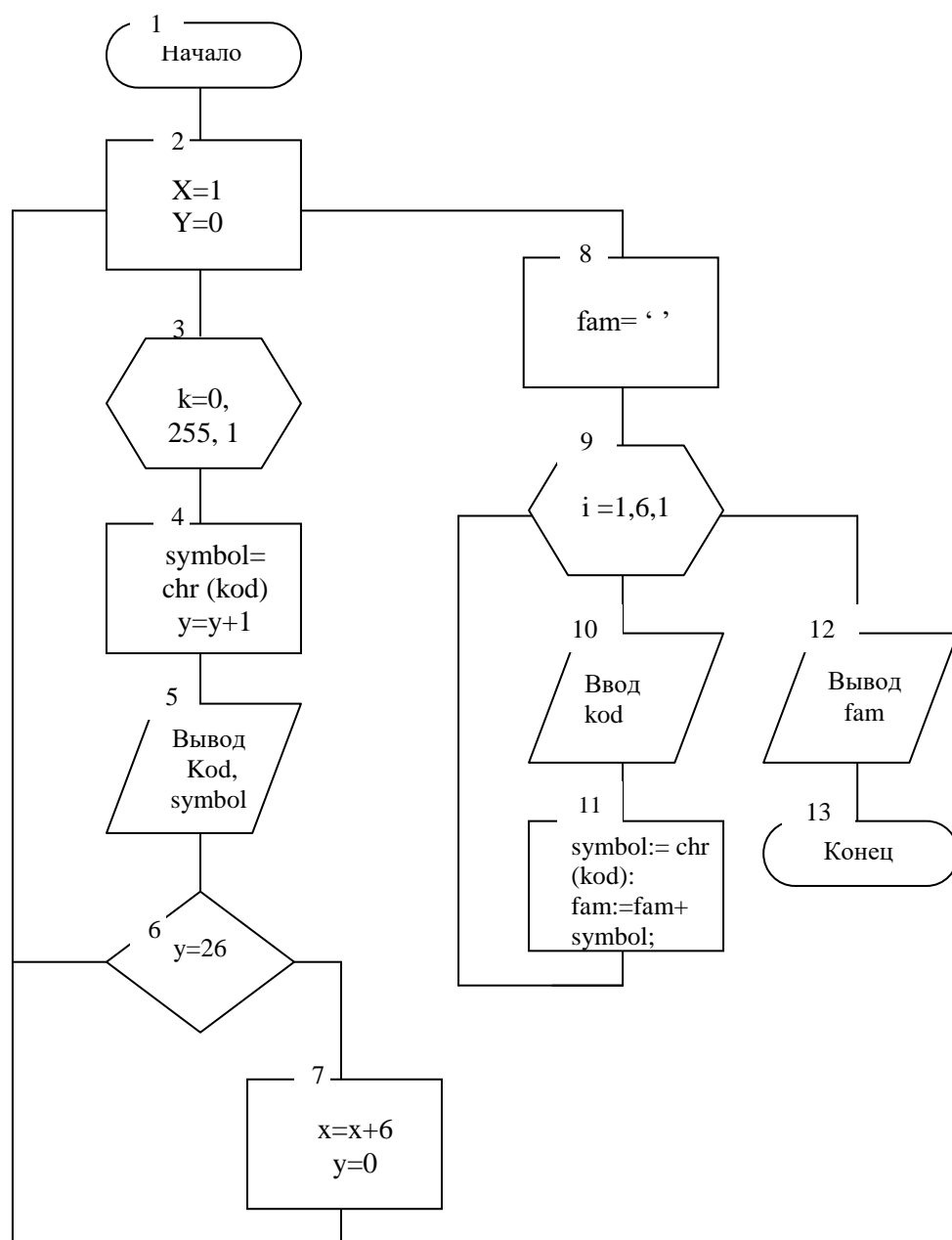
## 2.3 Реакция ЭВМ

0 =
...
20=¶
21=§
...
34=”
35=#
36=\$
...
40=(
41=)
42=*
43=+
...
49=1
50=2
51=3
...
75=K
76=L
...
84=T
85=U
...
128=A
129=Б
130=B
131=Г
132=Д

### 3 Пример формирования таблицы кодов и вывода на экран фамилии из найденных кодов

Составить блок-схему алгоритма и программу вывода на экран таблицы кодов в матричном виде, каждый столбик которой содержит 26 элементов. Сформировать из найденных кодов фамилию (Иванова) и вывести его на экран.

#### 3.1 Блок-схема алгоритма



### 3.2 Текст программы

```
Program UIPA;
uses crt;
var
  i:integer;
  x,y,kod:byte;
  symbol:char;
  fam: string [7];
begin
  clrscr;
  {формирования таблицы кодов}
  x:=1;y:=0;
  for kod:=0 to 255 do
    begin
      symbol:=chr(kod);
      y:=y+1;
      gotoxy(x,y);
      write (kod,'=',symbol);
      if y=26 then
        begin
          x:=x+6;
          y:=0;
        end;
    end;
  readln;
  clrscr;
  {формирование фамилии из заданных кодов букв}
  writeln('введите коды символов фамилии');
  fam:= ' ';
  for i:=1 to 6 do
    begin
      readln (kod);
      symbol:=chr (kod);
      fam:=fam+symbol;
    end;
  writeln (fam);
  readln;
end.
```

Таблица 1. Коды символов

код	Сим-вол	код	Сим-вол	код	Сим-вол	код	Сим-вол	код	Сим-вол	код	Сим-вол	код	Сим-вол
104	H	130	B	156	Ь	182		208	ll	234	Ъ		
105	I	131	Г	157	Э	183	П	209	т	235	Ы		
106	J	132	Д	158	Ю	184	Э	210	П	236	Ь		
107	K	133	Е	159	Я	185		211	ll	237	Э		
108	L	134	Ж	160	а	186		212	ll	238	Ю		
109	M	135	З	161	б	187	П	213	П	239	Я		
110	N	136	И	162	в	188		214	П	240	Ё		
111	о	137	Й	163	г	189	П	215		241	ё		
112	р	138	К	164	д	190		216		242	Є		
113	q	139	Л	165	е	191	П	217	П	243	є		
114	г	140	М	166	ж	192	П	218	П	244	İ		
115	s	141	Н	167	з	193	П	219	■	245	ï		
116	t	142	О	168	и	194	П	220	■	246	ÿ		
117	u	143	П	169	й	195	П	221	■	247	ÿ		
118	v	144	Р	170	к	196	—	222	■	248	°		
119	w	145	С	171	л	197	П	223	■	249	·		
120	x	146	Т	172	м	198	П	224	р	250	·		
121	y	147	У	173	н	199		225	с	251	√		
122	z	148	Ф	174	о	200	ll	226	т	252	№		
123	{	149	Х	175	п	201	П	227	у	253	ø		
124		150	Ц	176	■	202	ll	228	ф	254	■		
125	}	151	Ч	177	■	203	П	229	х	255			
126	~	152	Ш	178	■	204		230	ц				
127	△	153	Щ	179		205	=	231	ч				
128	A	154	Ъ	180		206		232	ш				
129	B	155	Ы	181		207		233	щ				

код	Сим-вол	код	Сим-вол	код	Сим-вол	код	Сим-вол
0		26	→	52	4	78	N
1	☺	27	←	53	5	79	O
2	☹	28	⬇	54	6	80	P
3	♥	29	↔	55	7	81	Q
4	♦	30	▲	56	8	82	R
5	♣	31	▼	57	9	83	S
6	♠	32		58	:	84	T
7		33	!	59	;	85	U
8		34	“	60	<	86	V
9	○	35	#	61	=	87	W
10		36	\$	62	>	88	X
11	♂	37	%	63	?	89	Y
12	♀	38	&	64	@	90	Z
13		39	‘	65	A	91	[
14	♫	40	(	66	B	92	\
15	☼	41	)	67	C	93	]
16	▲	42	*	68	D	94	^
17	▼	43	+	69	E	95	-
18	↑	44	,	70	F	96	,
19	!!	45	_	71	G	97	a
20	¶	46	.	72	H	98	b
21	§	47	/	73	I	99	c
22		48	0	74	J	100	d
23	↕	49	1	75	K	101	e
24	↑	50	2	76	L	102	f
25	↓	51	3	77	M	103	g

### 3.3 Реакция ЭВМ (фрагмент программы формирования фамилии)

Введите коды символов фамилии  
136  
162  
160  
173  
174  
162  
160  
  
Иванова

### 4 Контрольные вопросы

1. Сколько символов можно закодировать с помощью 0,5 байт?
2. Какие типы символов вы знаете?
3. Почему символам украинского алфавита соответствуют числа, больше числа, соответствующие символам латинского алфавита?
4. Объясните схему алгоритма для вывода символов в табличном виде (раздел 3).
5. Приведите пример использования функции ord (x).
6. Объясните схему алгоритма для вывода символов и их кодов на экран (раздел 2).
7. Объясните программу для вывода фамилии на экран (раздел 3).

8. Приведите пример использования функции chr (x).
9. Тип byte, для чего он используется?
10. Для чего используется оператор ClrScr?
11. Какие операции можно выполнять над строками символов?
12. Что означает это выражение: s = chr (k)?
13. Как вывести коды и соответствующие символы в центре экрана?
14. С помощью какой функции задаются координаты положения начала текста на экране?
15. Что нужно добавить или изменить в программе, чтобы записать ее результат в виде матрицы (раздел 2)?
16. Для чего в программе (раздел 3) используется оператор условного перехода?
17. Как задержать результат на экране?
18. Как описать в программе переменную  $\alpha$  типа Char?

## 5 Задание к лабораторной работе

1. Составить блок-схему алгоритма и программу для вывода всех символов и их кодов на экран (255 символов).
2. С помощью программы преобразования кодов в символы, определить коды букв, из которых состоит ваша фамилия.
3. Составить блок-схему алгоритма и программу для формирования из найденных кодов своей фамилии с последующим выводом его на экран.
4. Модифицировать схему алгоритма и программу задания 1 с целью вывода на экран символов и их кодов в табличном виде. Каждый столбец таблицы должен составлять (25-n) символов  $n \leq 15$  и (40-n) символов  $n \geq 15$ . Где n - номер варианта.
5. Модифицировать блок-схему алгоритма и программу задачи 2 с целью вывода на экран заданного текста номером варианта.

*Таблица 2. Варианты задания*

Вариант	Текст	Вариант	Текст
<b>1</b>	Переменную можно превратить в код	<b>16</b>	Текстовый файл представляет файловая переменная
<b>2</b>	Для хранения данных используют команды	<b>17</b>	Программа формирования файлов с



	или операторы языка программирования		последовательным доступом
<b>3</b>	Текстовые файлы возможно моделировать	<b>18</b>	В процессе выполнения программы временно выделять
<b>4</b>	Символы представляют собой стандартный тип данных	<b>19</b>	С помощью динамических переменных создают списки
<b>5</b>	Каждый символ имеет свой внутренний код	<b>20</b>	Каждый символ кодируется числом
<b>6</b>	Любой символ имеет предыдущий символ	<b>21</b>	Символами русского алфавита соответствуют большие числа
<b>7</b>	Тип string используется для обработки текстов	<b>22</b>	Переменные можно использовать при конструировании условий
<b>8</b>	Переменным типа char можно присваивать любые символы	<b>23</b>	При работе данные хранятся
<b>9</b>	Динамические переменные используются для экономии памяти	<b>24</b>	последовательные файлы состоят из элементов различной длины
<b>10</b>	Различают два типа файлов: последовательного и произвольного доступов	<b>25</b>	Файлы произвольного доступа состоят из однотипных элементов
<b>11</b>	Типы данных представляются совокупностью нулей и единиц	<b>26</b>	Можно эту информацию хранить в одном последовательном текстовом файле
<b>12</b>	Каждому байту соответствует символ	<b>27</b>	Объекты списка называются элементами списка
<b>13</b>	Файлом называется программа или данные	<b>28</b>	Над динамическими переменными можно делать любые математические и логические действия
<b>14</b>	Программист должен четко представлять	<b>29</b>	Для решения задания употребляют процедуру
<b>15</b>	Можно элемент	<b>30</b>	Создание модуля требует

	массива найти по его порядковому номеру		определенной его организации
--	--	--	---------------------------------