



IX ОЛИМПИАДА ПО ИНФОРМАТИКЕ И КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

Задача 1. Генерация пароля. (5 баллов)

Агент секретной службы Смит – начальник отдела по сбору секретных и компрометирующих материалов по западному региону, в котором для хранения информации создан специальный закрытый ресурс *PiggyLeaks.ru*, доступ к которому осуществляется с использованием пароля. Каждому сотруднику отдела выдается свой уникальный пароль доступа к материалам. При генерации паролей агент Смит ввел следующие ограничения:

- пароль состоит из 10 цифр, каждая из которых может принимать значение от 1 до 6 включительно;
- сумма любых трех соседних цифр в пароле равна 10.

Помогите агенту Смицу написать программу по генерации паролей для своих сотрудников. Сколько всего сотрудников может работать у него в отделе?

Задача 2. Алгоритм. (3 балла)

Квадратную матрицу размером n на n заполнили целыми числами по алгоритму, представленному на блок-схеме (см. рис. 1). При обращении к элементам массива переменная i обозначает номер строки, а переменная j – номер столбца. Индексация начинается с единицы. Найдите минимальное целое положительное значение m , при котором сумма элементов в матрице будет равняться 161, если $n = 13$?

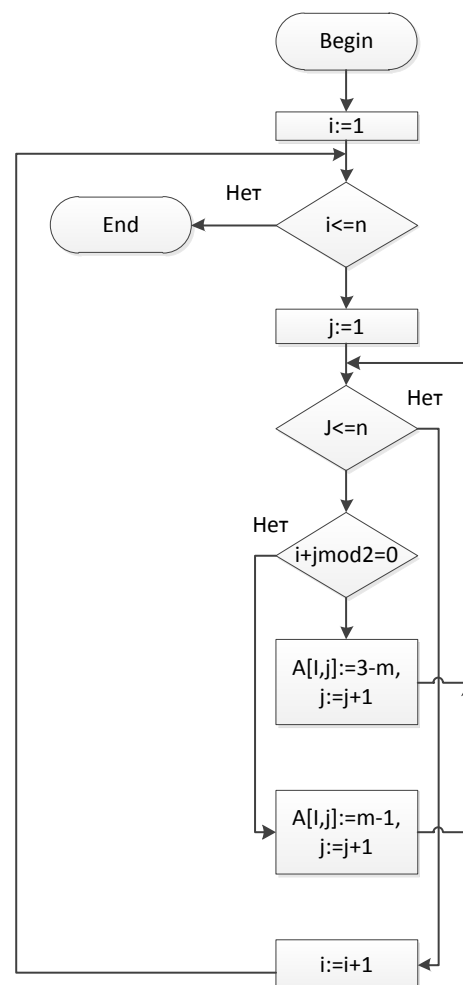


Рис. 1. Блок-схема алгоритма обработки матрицы

Задача 3. Коллизия. (4 балла)

В результате анализа подсистемы защиты удалось выяснить, что подтверждение имени пользователя, работающего за компьютером, осуществляется посредством паролей. Для реализации проверки введенный пароль подвергается преобразованию при помощи функции, исходный код которой также был восстановлен и представлен на языке C (см. рис. 2). Результат работы функции сверяется с эталонами, хранимыми в базе данных, для принятия решения о том верен пароль или нет. Если известно, что пароль «ЦЕЛОСТНОСТЬ» верен, то можно ли утверждать, что существуют пароли, отличные от указанного, которые пройдут описанную проверку? Если да, то приведите пример такого пароля.

```
int HASH(char *text)
{
    int k=0, H;
    char temp[4], letter = 'F';
    for(; (k<strlen(text)) && (k<4); ++k) temp[k]=text[k];
    if (k < 4) for(;k<4;k++) temp[k]= letter;
    H = ((temp[0]-temp[1])&7)*8;
    H += ((temp[3]-temp[2])&7);
    return H;
}
```

Рис. 2. Исходный код

Фрагмент таблицы ASCII

hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	“	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	:	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
8	A	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
9	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я

Задача 4. Волшебная гора. (3 балла)

На волшебной горе расположена система озер и рек. Верхнее озеро образуется из тающего ледника. Из каждого озера вытекает ровно две реки, а впадает одна (за исключением верхнего). Первым на горе возле верхнего озера поселился гном Тим. Каждый новый обитатель горы строил свой дом как можно ближе к Тиму на одном из озер. На каждом озере селился только один гном. Сейчас на горе живут 2014 гномов. Какое количество рек надо проплыть Тиму, чтобы попасть в гости к последнему поселившемуся гному.

Задача 5. Лягушка. (4 балла)

Что делает программа (см. Рис.3)?

```
Uses crt;
Const z=10;
hhh=10; y= 2*
hhh+ 1;s= {} hhh- z+
1; Type vec={a -1 ab
x} {} array[s..z] {} of {a ,v
v+1} integer; {} var a,b: {}
vec;v,f:{hh} {xy} {ab}
integer; x:Boolean; {}
Begin clrscr;{a,b,x} {xy}
{ab} Randomize; {}
{hh} {} write( '[]=' );for v:=
s to z do{v x} begin//
f:= random (y);a[v]:= hhh-f+s-1;
write(a[v]:3);End; writeln;b:= a;Repeat {}
x:={x} true;for{a }v:=s to z- 1 do if {}
b[ v] >b[v+1]then Begin{z} f:=b[v];b[
v] := b[ v+1];b[v+1]:=f; {} x:=false;
End; until x;{ab }write(
'[]=');for v:=s to
{v-1}z
do{v+1-2*x}
write{x[a-2]
}(b[v]:3
); End.
```

Рис. 3. Код программы.