

VOORBLAD SCHRIFTELIJKE TOETSEN

OPLEIDING	: Mechatronica
TOETSCODE	: INPROG1-T1
GROEP	: MeP1
TOETSDATUM	: 10 november 2011
TIJD	: 09.00 – 10.30 uur 11.00 – 12.30 uur 13.00 – 14.30 uur 15.00 – 16.30 uur Anders: uur (ivm duale opleidingen)
AANTAL PAGINA'S (incl. dit voorblad)	: 5
DEZE TOETS BESTAAT UIT	: 4 open vragen (aantal) geen MC-vragen (aantal)
GEBRUIK HULPMIDDELEN	: JA/NEE
TOEGESTANE HULPMIDDELEN	: Boek over C programmeren en zelfgeschreven aantekeningen
OVERIGE OPMERKINGEN	: Elk willekeurig boek over C programmeren is toegestaan
OPSTELLER VAN DEZE TOETS	: J. van Peski
NAAM 2^E LEZER	: J. Lambers

Wegens het ontbreken van een compiler die reageert op typefouten zal hier tijdens het beoordelen niet teveel op worden gelet. Er wordt voornamelijk gelet op structuur, netheid van het programma (leesbaarheid) en vooral correctheid van het algoritme.

1. Schrijf een programma dat: 10 punten
Van de onderstaand gegeven, eendimensionale, array de inhoud omkeert.

```
int getal [] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
```

na afloop van het programma bevat de array de volgende inhoud:
{ 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }

2. Schrijf een programma dat: 25 punten
Van de onderstaand gegeven, tweedimensionale, array de inhoud horizontaal omkeert.

```
int getal [][][10] = { { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 },  
                      { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 },  
                      { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 },  
                      { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 },  
                      { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } };
```

na afloop van het programma bevat de array de volgende inhoud:
{ { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }
 { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }
 { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }
 { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 }
 { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1 } }

3. Schrijf een programma dat: 25 punten
Karakters van het toetsenbord leest.
Van alle ingelezen karakters worden van de **cijfers** de individuele getalwaarden bij elkaar opgeteld waarna het totaal wordt afgedrukt.

Voorbeeld:

Indien de onderstaande regel wordt ingetypt dan zal het resultaat (1 + 1 + 5 + 2 = 9) bedragen:

Er zwemt 1 eend in de vijver, 15 ganzen lopen in de tuin en 2 varkens staan in de stal.

LET OP, de waarden van de karakters (cijfers) die je inleest is ASCII code !

4. Schrijf een programma dat:

30 punten

Het volgende bestand inleest: **c:/temp/hartscan.txt**

Het bestand **c:/temp/hartscan.txt** bevat de volgende inhoud:

```
1 3 1 7 1 1 39 1 3 1 5
1 1 7 1 1 33 1 9 1 1 1
5 3 4 6 85 2 4 4 1 1 43
1 1 6 47 4 4 4 3 8 23 1
3 1 45 4 4 4 5 4 2 31 7
1 5 42 4 4 7 4 4 54 5 1
1 1 30 4 9 4 2 44 4 1 9
5 7 25 3 6 4 45 8 4 5 1
1 1 3 21 41 30 4 6 4 1 3
1 5 1 3 1 1 3 1 1 9 1
7 1 3 1 1 9 1 1 7 1 5
```

Het resultaat in het uitvoerbestand: **c:/temp/contrast.txt** ziet er als volgt uit:

```
-----X-----
-----X-----
----X-----X
---X-----X-
--X-----X-
--X-----X--
--X-----X---
--X---X-----
----XX-----
-----
-----
```

HINT: Het omslagpunt waarbij een X of een - wordt opgeslagen ligt bij de waarde 20 !

Bijlage 1

isdigit()

Syntax:

```
#include <ctype.h>
int isdigit( int ch );
```

Description:

The function `isdigit()` returns non-zero if its argument is a digit between 0 and 9. Otherwise, zero is returned.

Example:

```
char c;
scanf( "%c", &c );
if( isdigit(c) )
    printf( "You entered the digit %c\n", c );
```

Bijlage 2

isalpha()

Syntax:

```
#include <ctype.h>
int isalpha( int ch );
```

Description:

The function `isalpha()` returns non-zero if its argument is a letter of the alphabet. Otherwise, zero is returned.

Example:

```
char c;
scanf( "%c", &c );
if( isalpha(c) )
    printf( "You entered a letter of the alphabet\n" );
```

Bijlage 3

ASCII Tabel

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	MUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.LookupTables.com