

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**ΕΠΛ 035: Δομές Δεδομένων και Αλγόριθμοι για Ηλεκτρολόγους Μηχανικούς και
Μηχανικούς Υπολογιστών**

Χειμερινό Εξάμηνο 2012

**ΑΣΚΗΣΗ 1
Συμβολοσειρές, Πίνακες, Δείκτες**

Διδάσκων Καθηγητής: Παναγιώτης Ανδρέου

**Ημερομηνία Υποβολής: 10/09/2012
Ημερομηνία Παράδοσης: 24/09/2012**

Υπεύθυνος Εργαστηρίου: Πέτρος Παναγή (petrosp@cs.ucy.ac.cy)

ΠΕΡΙΓΡΑΦΗ

Σε αυτή την άσκηση καλείστε να υλοποιήσετε διάφορες βοηθητικές συναρτήσεις για συμβολοσειρές (strings). Όλα τα strings θα έχουν μέγιστο μέγεθος 50 χαρακτήρες. Το πρόγραμμά σας θα πρέπει να παρέχει τις ακόλουθες λειτουργίες:

1. Convert to Lowercase with Table
2. Convert to Lowercase with Pointers
3. Convert to Uppercase
4. Find character in String
5. Replace character with other character
6. Reverse String
7. Check if string starts with another string
8. Check if string ends with another string
9. EXIT

Σε κάθε επιλογή, ο χρήστης θα έχει την ευχέρεια να επιλέξει μία από τις συναρτήσεις και ακολούθως να εισαγάγει τις παραμέτρους της συνάρτησης. Για παράδειγμα, αν ο χρήστης επιλέξει από το μενού τη συνάρτηση 1, το πρόγραμμα θα ζητήσει την παράμετρο string που θα μετατραπεί (π.χ., "TEST") και μετά την επεξεργασία θα τυπώσει το επεξεργασμένο string στην οθόνη (δηλ. "test").

Η κάθε επιλογή θα πρέπει να υλοποιηθεί σαν μία ξεχωριστή συνάρτηση. Οι λεπτομέρειες, περιορισμοί της κάθε συνάρτησης περιγράφονται πιο κάτω:

1. Convert to Lowercase with Tables

Ο χρήστης δίνει κάποιο string <str> ως τιμή εισόδου. Το πρόγραμμα στη συνέχεια μετατρέπει όλα τα κεφαλαία γράμματα (upper-case) σε πεζά γράμματα (lower-case) επί τόπου, δηλ., αλλάζει το ίδιο το string <str> και συνεπώς δεν χρειάζεται η χρήση δευτερεύων string μεταβλητής. Σημειώστε ότι η μετατροπή πρέπει να γίνεται μόνο για χαρακτήρες του λατινικού αλφαβήτου; οι υπόλοιποι χαρακτήρες πρέπει να παραμένουν ως έχουν.

ΠΕΡΙΟΡΙΣΜΟΙ: Επιτρέπεται η χρήση πινάκων. Δεν επιτρέπεται η χρήση δεικτών και αριθμητικής δεικτών

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=TEST123 τυπώνει <str>=test123
- <str>=TeST 123 τυπώνει <str>=test 123
- <str>=test123 τυπώνει <str>=test123

Πρότυπο συνάρτησης: `void lower_T(char str[])`

2. Convert to Lowercase with Pointers

Ο χρήστης δίνει κάποιο string <str> ως τιμή εισόδου. Το πρόγραμμα στη συνέχεια μετατρέπει όλα τα κεφαλαία γράμματα (upper-case) σε πεζά γράμματα (lower-case) επί τόπου, δηλ., αλλάζει το ίδιο το string <str> και συνεπώς δεν χρειάζεται η χρήση δευτερεύων string μεταβλητής. Σημειώστε ότι η μετατροπή πρέπει να γίνεται μόνο για χαρακτήρες του λατινικού αλφαβήτου; οι υπόλοιποι χαρακτήρες πρέπει να παραμένουν ως έχουν.

ΠΕΡΙΟΡΙΣΜΟΙ: Επιτρέπεται η χρήση δεικτών και αριθμητικής δεικτών. Δεν επιτρέπεται η χρήση πινάκων.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=TEST123 τυπώνει <str>=test123
- <str>=TeST 123 τυπώνει <str>=test 123
- <str>=test123 τυπώνει <str>=test123

Πρότυπο συνάρτησης: `void lower_P(char* str)`

3. Convert to Uppercase

Ο χρήστης δίνει κάποιο string <str> ως τιμή εισόδου. Το πρόγραμμα στη συνέχεια μετατρέπει όλα τα πεζά γράμματα (lower-case) σε κεφαλαία γράμματα (upper-case) επί τόπου, δηλ., αλλάζει το ίδιο το string <str> και συνεπώς δεν χρειάζεται η χρήση δευτερεύων string μεταβλητής. Σημειώστε ότι η μετατροπή πρέπει να γίνεται μόνο για χαρακτήρες του λατινικού αλφαβήτου; οι υπόλοιποι χαρακτήρες πρέπει να παραμένουν ως έχουν.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=TEST123 τυπώνει <str>=TEST123
- <str>=TeST 123 τυπώνει <str>=TEST 123
- <str>=test123 τυπώνει <str>=TEST123

Πρότυπο συνάρτησης: `void upper(char* str)`

4. Find character in String

Ο χρήστης δίνει κάποιο string <str> και ένα χαρακτήρα <c> ως τιμές εισόδου. Το πρόγραμμα στη συνέχεια ψάχνει να βρει το χαρακτήρα <c> μέσα στο string <str>. Αν ο χαρακτήρας <c> βρεθεί μέσα στο string <str> επιστρέφεται η θέση του, αλλιώς επιστρέφεται η τιμή -1.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=This is a test, <c>=a επιστρέφει <int> 8
- <str>=This is a test, <c>=k επιστρέφει <int> -1
- <str>=This is a test, <c>=T επιστρέφει <int> 0

Πρότυπο συνάρτησης: `int findChar(char* str, char c)`

5. Replace character with other character in String

Ο χρήστης δίνει κάποιο string <str>, ένα χαρακτήρα <oldChar> και ένα χαρακτήρα <newChar> ως τιμές εισόδου. Το πρόγραμμα στη συνέχεια ψάχνει και αντικαθιστά κάθε χαρακτήρα <oldChar> μέσα στο string <str> με το χαρακτήρα <newChar> επί τόπου.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=This is, <oldChar>=T, <newChar>=t, τυπώνει <str>=this is
- <str>=This is, <oldChar>=i, <newChar>=-, τυπώνει <str>=th-s -s
- <str>=This is, <oldChar>=k, <newChar>=f, τυπώνει <str>=This is

Πρότυπο συνάρτησης: `void replaceChar(char* str, char oldChar, char newChar)`

6. Reverse String

Ο χρήστης δίνει κάποιο string <str> ως τιμή εισόδου. Το πρόγραμμα στη συνέχεια αντιστρέφει το string <str> και το επιστρέφει.

ΠΕΡΙΟΡΙΣΜΟΙ: Η μετατροπή δεν επηρεάζει την παράμετρο string <str>.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=This is επιστρέφει <char*>=si sihT
- <str>=android επιστρέφει <char*>=diordna
- <str>=abra kedavra επιστρέφει <char*>=arvadek arba

Πρότυπο συνάρτησης: `char* reverse(char* str)`

7. Check if string starts with another string

Ο χρήστης δίνει κάποιο string <str> και ένα string <check>. Το πρόγραμμα στη συνέχεια ψάχνει να βρει αν το string <check> είναι η αρχή του string <str>. Αν ισχύει τότε επιστρέφει TRUE αλλιώς FALSE.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=This is a test, <check>=This is επιστρέφει <bool> true
- <str>=This is a test, <check>=this επιστρέφει <bool> false
- <str>=This is a test, <check>="" επιστρέφει <bool> true

Πρότυπο συνάρτησης: `bool startsWith(char* str, char* check)`

8. Check if string ends with another string

Ο χρήστης δίνει κάποιο string <str> και ένα string <check>. Το πρόγραμμα στη συνέχεια ψάχνει να βρει αν το string <check> είναι το τέλος του string <str>. Αν ισχύει τότε επιστρέφει TRUE αλλιώς FALSE.

ΠΑΡΑΔΕΙΓΜΑΤΑ:

- <str>=This is a test, <check>=is a test επιστρέφει <bool> true
- <str>=This is a test, <check>=TEST επιστρέφει <bool> false
- <str>=This is a test, <check>="" επιστρέφει <bool> true

Πρότυπο συνάρτησης: `bool endsWith (char* str, char* check)`

ΧΡΗΣΙΜΕΣ ΠΛΗΡΟΦΟΡΙΕΣ ΚΩΔΙΚΑ

- Δήλωση Μέγιστου μεγέθους string:
`#define MAX 50`
- Παράδειγμα εισαγωγής string από το πληκτρολόγιο:
`printf("Enter input string:");
fgets(str, MAX, stdin);
fflush(stdin);`
- Ο Πίνακας ASCII (για συναρτήσεις 1-3)

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

ΑΞΙΟΛΟΓΗΣΗ

Για την αξιολόγηση του προγράμματος σας, θα ληφθούν οι πιο κάτω παράμετροι:

- **Ορθότητα:** ο πρόγραμμα σας πρέπει τρέχει ορθά για οποιαδήποτε είσοδο
- **Κατανοητά σχόλια:** Γράφετε κατανοητά σχόλια που να εξηγούν την λειτουργία της κάθε δομής/συνάρτησης.

ΟΔΗΓΙΕΣ

- Οι άσκηση σας θα πρέπει να παραδοθεί σε ηλεκτρονική μορφή μέσω email στον υπεύθυνο βοηθό σαν ένα αρχείο (.c) ή/και ένα αρχείο (.h), που να περιλαμβάνει τον κώδικά σας.
- Το αρχείο (.c) θα ονομάζεται με τον εξής τρόπο:
erl035.ex<αριθμός άσκησης>.<ταυτότητα>.c (π.χ., erl035.ex1.123456.c)
- Στην περίπτωση χρήσης βιβλιοθήκης (.h) τότε το αρχείο θα ονομάζεται με τον εξής τρόπο:
erl035.ex<αριθμός άσκησης>.<ταυτότητα>.h (π.χ., erl035.ex1.123456.h)
- Βεβαιωθείτε ότι τα προγράμματα σας είναι ορθά και τρέχουν.
- Τα αρχεία σας πρέπει να μεταγλωττίζονται ορθά με τον μεταγλωττιστή (compiler) gcc του εργαστηρίου UNIX. Παράδειγμα μεταγλώττισης:
gcc erl035.ex1.123456.c -o a.out
- Στην αρχή του κάθε αρχείου θα πρέπει να αναγράφεται η ταυτότητά σας σε σχόλια
- Επίσης θα πρέπει να παραδώσετε μία αναφορά, σε έντυπη μορφή η οποία θα περιέχει τον κώδικα σας και ένα αντιπροσωπευτικό παράδειγμα εκτέλεσης του μαζί με τα σχόλια που ζητούνται πιο πάνω.
- Στην αναφορά σας θα πρέπει να αναφέρετε συγκεκριμένα ποιες συναρτήσεις δουλεύουν και ποιες όχι.
- Μη τήρηση των ημερομηνιών παράδοσης των εργασιών συνεπάγεται τις ανάλογες βαθμολογικές επιπτώσεις (μέχρι τον μηδενισμό της εργασίας).
- Οι προγραμματιστικές ασκήσεις θα ελέγχονται από ειδικό πρόγραμμα για την ανίχνευση των αντιγραφών. Οι αντιγραμμένες εργασίες θα μηδενίζονται και για τους αντιγραφείς θα εφαρμόζονται οι κανόνες τού Πανεπιστημίου. Αποφύγετε λοιπόν την αντιγραφή προγραμμάτων από άλλους συναδέλφους σας, διότι έτσι εκτίθετε και αυτούς και τον εαυτό σας στον κίνδυνο μηδενισμού και πειθαρχικής δίωξης.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ!!!