

# C Ավազան C 02

<ակիրճ. այս փաստաթուղթը 42 դպրոցի C ավազանի C 02 մոդուլի նյութն t:

# Ցանկ

1	anigninguh	2
II	Նախաբան	4
III	Առաջադրանը 00 ։ ft_strcpy	5
IV	Առաջադրանը 01 ։ ft_strncpy	6
V	Առաջադրանը 02 ։ ft_str_is_alpha	7
VI	Առաջադրանը 03 ։ ft_str_is_numeric	8
VII	Առաջադրանը 04 ։ ft_str_is_lowercase	9
VIII	Առաջադրանը 05 ։ ft_str_is_uppercase	10
IX	Առաջադրանք 06 ։ ft_str_is_printable	11
X	Առաջադրանը 07 ։ ft_strupcase	12
XI	Առաջադրանը 08 ։ ft_strlowcase	13
XII	Առաջադրանը 09 ։ ft_strcapitalize	14
XIII	Առաջադրանը 10 ։ ft_strlcpy	15
XIV	Առաջադրանք 11 ։ ft_putstr_non_printable	16
XV	Առաջադրանք 12 ։ ft_print_memory	17
XVI	Հանձնում և ընկերն ընկերոջը ստուգում	19

# Գլուխ I

#### Յուցումներ

- Այս էջը ձեր միակ ուղեցույցն է։ Պտտվող խոսակցություններին ուշադրություն մի՛ դարձրեք։
- Չգուշացում. մինչ առաջադրանքները հանձնելը նորից ստուգե՛ք նյութը։ Ցանկացած պահի այս փաստաթուղթը կարող է փոփոխվել։
- Ուշադրություն դարձրե՛ք ձեր ֆայլերի և պահոցների թույլտվություններին։
- Բոլոր առաջադրանքները կատարելիս անհրաժեշտ է հետևել հանձման ընթացակարգին։
- Ձեր առաջադրանքները կստուգվեն ձեր դասընկերների կողմից։
- Քացի դրանից, ձեր առաջադրանքները կստուգվեն և կգնահատվեն Moulinette կոչվող ծրագրով։
- Moulinette-ը գնահատելիս շատ բծախնդիր է ու խիստ։ Այն ամբողջովին ավտոմատացված է, և գնահատման հարցում նրա հետ անհնար է բանակցել։ Այսպիսով, տհաճ անակնկալներից խուսափելու համար առաջադրանքները պետք է կատարվեն հնարավորինս անթերի։
- Moulinette-ն այնքան էլ լայնախոհ չէ։ Այն չի էլ փորձի հասկանալ ձեր կոդը, եթե վերջինս չի համապատասխանում Norm-ին։
- Մուլինետի աշխատանքը հիմնված է norminette կոչվող ծրագրի վրա, որը ստուգում է, թե արդյոք ձեր ֆայլերը համապատասխանում են Norm-ին։ Կարճ ասած, norminette-ի ստուգման թեստը չանցած աշխատանքը չի ընդունվի։
- Առաջադրանքները դասավորված են ըստ բարդության աստիճանի` ամենապարզից ամենաբարդը։ Հաջողությամբ կատարված բարդ առաջադրանքները հաշվի չեն առնվի, եթե պարզ առաջադրանքներից որևէ մեկը լիարժեք չի աշխատում։
- Արգելված ֆունկցիաների կիրառումը համարվում է խարդախություն։ Խարդախությունը պատժվում է -42-ով, և այս գնահատականը քննարկման ենթակա չէ։

C Ավազան

C 02

- main () ֆունկցիա պետք է հանձնել միայն այն դեպքում, եթե պահանջվի գրել ծրագիր։
- Moulinette-ը կազմարկում է այս դրոշակների օգնությամբ` -Wall -Wextra -Werror, և գործածում է cc:
- Եթե ձեր ծրագիրը չկազմարկվի, կստանաք 0։
- Նյութում նշված ֆայլից բացի ձեր պահոցում <u>հավելյալ</u> ֆայլեր չպետք է լինեն։
- Հարցեր կա՞ն։ Դիմե՛ք աջ կողմում նստած դասընկերոջը։ Կամ Էլ դիմե՛ք ձախ կողմինին։
- Ձեր ուղեցույցների անուններն են Google/ man/ internet/ ...
- Կարող եք օգտվել նաև ներքնացանցի ֆորումի «C Piscine» hատվածից կամ slack Piscine-ից։
- Մանրակրկիտ ուսումնասիրե՛ք օրինակները։ Շատ հնարավոր է, որ դրանք պահանջեն նյութում հստակորեն չնշված մանրամասներ։
- Դե՜, ձեզ տեսնենք։ Հանուն Օդինի, հանուն Արամազդի։ Ուղեղներդ ի գո՜րծ։



Norminette-ը պետք է գործարկվի -R CheckForbiddenSourceHeader դրոշակով։ Այն նաև անհրաժեշտ է Moulinette-ի աշխատանքի համար։

# Գլուխ II Նախաբան

Ահա մի հատված «*Սիլիկոնյան հովիտ*» սերիալից։

- Լսի՜ր, ինչու՞ ուղղակի չգործածել Vim-ը Emacs-ի փոխարեն (ԾԻԾԱՂ)։

- Ես հենց այդպես էլ անում եմ՝ Vim-ր Emacs-ի փոխարեն։

- O´, տե՛ր Աստված, փրկի՛ր մեզ։ Գիտե՞ս ինչ, ինձ չի թվում, որ սա լավ միտք է։ Կներես։ Լսի՛ր, մենք պատրաստվում ենք երեխաներ բերել էս աշխարհ ու էսպիսի մեծ խնդի՞ր թողնել իրենց ուսերի՞ն։ Էնքան էլ արդար չի նրանց նկատմամբ, չէ՞։

- Երեխանե՞ր, մենք իրար հետ չենք էլ քնել։

- Ու գիտե՞ս ինչ, այդպիսի բան երբեք էլ չի լինի, ես երբեք չեմ լինի մեկի հետ, ով գերադասում է բացատները tab-երից։

- Ռիչա՜րդ։ (Անընդհատ սեղմում է բացատի կոճակը)։

- Ա՛յ քեզ բան, լավ, հաջողություն։

- Մի tab-ը ութ բացատ է խնայում։ (ԴՈԻՌԸ ՓԱԿՎՈԻՄ Է) - (ՇՐԸ՜ԽԿ)

․․․ (Ռիչարդը տնքտնքում է)

- Աստվա՜ծ իմ, Ռի՜չարդ, ի՞նչ եղավ։

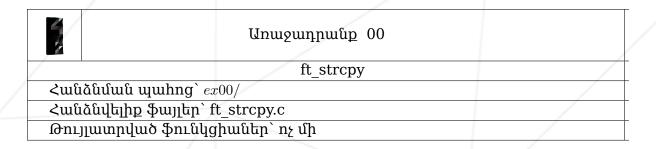
- Ուղղակի փորձեցի միանգամից ութ աստիճան իջնել, բայց ես լավ եմ, մի՛ անհանգստացիր։

- Ցտեսություն, Ռիչա՛րդ։

- Չուտ տեսակետս էի պաշտպանում։

Հուսով ենք` այս առաջադրանքը կատարելիս ձեզ չեն ստիպի emacs ու բացատի կոճակ գործածել։

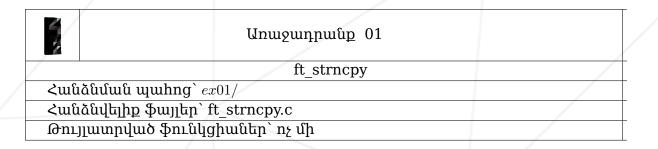
# Գլուխ III Առաջադրանը 00 : ft\_strcpy



- Վերարտադրել strcpy (man strcpy) ֆունկցիայի վարքը։
- Նախատիպը պետք է լինի այսպիսին`

char \*ft\_strcpy(char \*dest, char \*src);

# Գլուխ IV Առաջադրանը 01 : ft\_strncpy

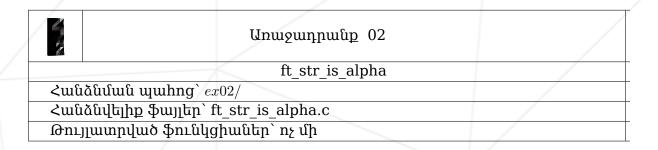


- Վերարտադրել strncpy (man strncpy) ֆունկցիայի վարքը։
- Նախատիպը պետք է լինի այսպիսին`

char \*ft\_strncpy(char \*dest, char \*src, unsigned int n);

#### Գլուխ V

## Առաջադրանը 02 ։ ft\_str\_is\_alpha



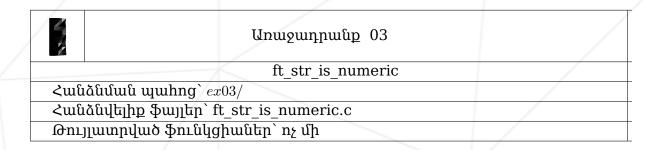
- Գրել ֆունկցիա, որը վերադարձնում է 1, եթե որպես պարամետր արված տողը պարունակում է միայն տառեր, և 0, եթե պարունակում է այլ նիշեր։
- Նախատիպը պետք է լինի այսպիսին`

int ft\_str\_is\_alpha(char \*str);

• Կվերադարձնի 1 եթե str-ն դատարկ է.

#### Գլուխ VI

## Առաջադրանք 03 ։ ft\_str\_is\_numeric



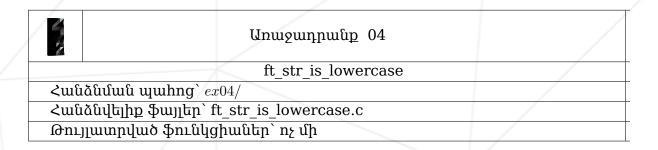
- Գրել ֆունկցիա, որը վերադարձնում է 1, եթե որպես պարամետր արված տողը պարունակում է միայն թվեր, և 0, եթե պարունակում է այլ նիշեր։
- Նախատիպը պետք է լինի այսպիսին`

int ft\_str\_is\_numeric(char \*str);

• Պետք է վերադարձնի 1, եթե str-ը դատարկ է։

#### Գլուխ VII

## Առաջադրանք 04 ։ ft\_str\_is\_lowercase



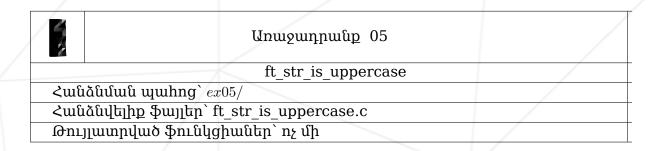
- Գրել ֆունկցիա որը վերադարձնում է 1, եթե որպես պարամետր արված տողը պարունակում է միայն փոքրատառեր, և 0 ,եթե պարունակում է այլ նիշեր։
- Նախատիպը պետք է լինի այսպիսին`

int ft\_str\_is\_lowercase(char \*str);

• Պետք է վերադարձնի 1 եթե str դատարկ է.

#### Գլուխ VIII

## Առաջադրանք 05 ։ ft\_str\_is\_uppercase



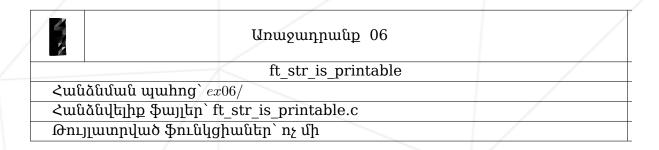
- Գրել ֆունկցիա որը վերադարձնում է 1, եթե որպես պարամետր տրված տողը պարունակում է միայն մեծատառեր, և 0, եթե պարունակում է այլ նիշեր։
- Նախատիպը պետք է լինի այսպիսին`

int ft\_str\_is\_uppercase(char \*str);

• Պետը է վերադարձնի 1, եթե str դատարկ է։

#### Գլուխ IX

## Առաջադրանք 06 ։ ft\_str\_is\_printable



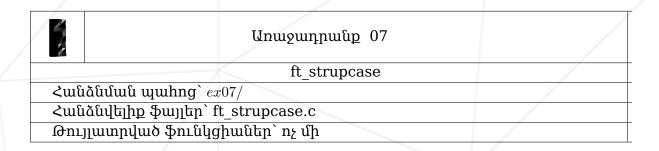
- Գրել ֆունկցիա որը վերադարձնում է 1, եթե որպես պարամետր արված տողը պարունակում է միայն արտածելի նիշեր, և 0, եթե պարունակում է այլ նիշեր։
- Նախատիպը պետք է լինի այսպիսին`

int ft\_str\_is\_printable(char \*str);

• Պետք է վերադարձնի 1, եթե str դատարկ է։

#### Գլուխ X

## Առաջադրանք 07 ։ ft\_strupcase



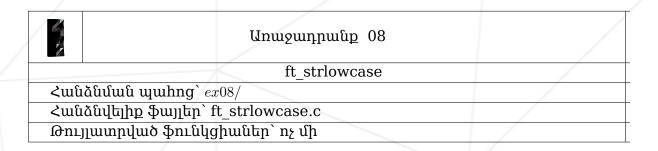
- Գրել ֆունկցիա, որը բոլոր տառերը դարձնում է մեծատառ։
- Նախատիպը պետք է լինի այսպիսին`

char \*ft\_strupcase(char \*str);

• Պետք է վերադարձնի str.

## Գլուխ XI

## Առաջադրանք 08 ։ ft\_strlowcase



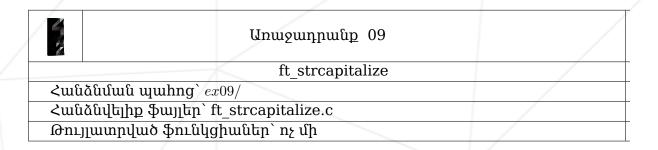
- Գրել ֆունկցիա, որը բոլոր տառերը դարձնում է փոքրատառ։
- Նախատիպը պետք է լինի այսպիսին`

char \*ft\_strlowcase(char \*str);

• Պետք է վերադարձնի str.

# Գլուխ XII

## Առաջադրանը 09 ։ ft\_strcapitalize



- Գրել ֆունկցիա, որը յուրաքանչյուր բառի առաջին տառը դարձնում է մեծատառ, իսկ մնացած տառերը` փոքրատառ։
- Բառը տառերից ու թվերից կազմված նշախումբ է։
- Նախատիպը պետք է լինի այսպիսին`

char \*ft\_strcapitalize(char \*str);

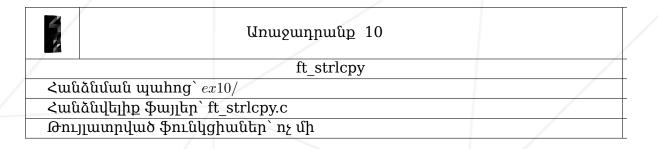
- Պետք է վերադարձնի str.
- Օրինակ՝

salut, comment tu vas ? 42mots quarante-deux; cinquante+et+un

• Կդառնա՝

Salut, Comment Tu Vas ? 42mots Quarante-Deux; Cinquante+Et+Un

# Գլուխ XIII Առաջադրանը 10 : ft\_strlcpy

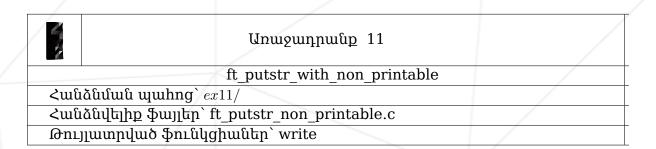


- Վերարտադրել strlcpy (man strlcpy) ֆունկցիայի վարքը։
- Նախատիպը պետք է լինի այսպիսին`

unsigned int ft\_strlcpy(char \*dest, char \*src, unsigned int size);

#### Գլուխ XIV

## Առաջադրանք 11 : ft\_putstr\_non\_printable



- Գրել ֆունկցիա, որը տողի նիշերը արտատպում է էկրանին։ Եթե տողում չտպվող նիշեր կան, դրանք կներկայացվեն տասվեցական համակարգով (փոքրատառերով), իսկ վերջում կդրվի "backslash"։
- Օրինակ՝

Coucou\ntu vas bien ?

• Ֆունկցիան պետք է ներկայացնի՝

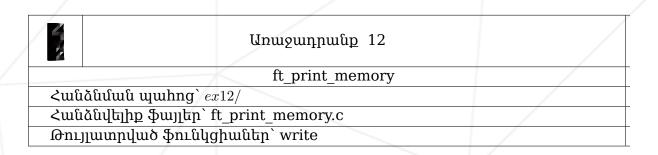
Coucou\Oatu vas bien ?

• Նախատիպը պետք է լինի այսպիսին`

ft\_putstr\_non\_printable(char \*str);

#### Գլուխ XV

## Առաջադրանք 12 ։ ft\_print\_memory



- Գրել ֆունկցիա, որը հիշողության տիրույթը արտապատկերում է Էկրանին։
  - <րշողության տիրույթի արտապատկերումը պետք է բաժանված լինի երեք սյուների, որոնք էլ իրարից պետք է առանձնացված լինեն բացատով։
    - o Տասնվեցական համակարգով արված հասցեի առաջին աողի առաջին նիշր , որին հաջորդում է ':'
    - Տասնվեցական համակարգով ներկայացված հատվածը, որտեղ ամեն երկու նիշից հետո բացատ է դրված, անհրաժեշտության դեպքում պետք է լրացվի բացատներով (տե՜ս ներքևի օրինակը)։
    - o Այդ հատվածը` ներկայացված է տպվող նիշերով։
- Եթե նիշը չարտածվող է, ապա այն կփոխարինվի կետով։
- Յուրաքանչյուր տող պետք է պարունակի տասնվեց նիշ։
- Եթե չափ/երկարությունը 0 է, ոչինչ չի արտապատկերվի։

C Ավազան C 02

• Օրինակ՝

```
$> ./ft_print_memory

00000010a161f40: 426f 6e6a 6f75 7220 6c65 7320 616d 696e Bonjour les amin

000000010a161f50: 6368 6573 090a 0963 2020 6573 7420 666f ches...c est fo

000000010a161f60: 7509 746f 7574 0963 6520 7175 206f 6e20 u.tout.ce qu on

000000010a161f70: 7065 7574 2066 6169 7265 2061 7665 6309 peut faire avec.

000000010a161f80: 0a09 7072 696e 745f 6d65 6d6f 7279 0a0a ..print_memory.

000000010a161f90: 0a09 6c6f 6c2e 6c6f 6c0a 2000 ..lol.lol. .

$> ./ft_print_memory | cat -te

0000000107ff9f40: 426f 6e6a 6f75 7220 6c65 7320 616d 696e Bonjour les amin$

0000000107ff9f50: 6368 6573 090a 0963 2020 6573 7420 666f ches...c est fo$

0000000107ff9f60: 7509 746f 7574 0963 6520 7175 206f 6e20 u.tout.ce qu on $

0000000107ff9f70: 7065 7574 2066 6169 7265 2061 7665 6309 peut faire avec.$

0000000107ff9f80: 0a09 7072 696e 745f 6d65 6d6f 7279 0a0a ..print_memory..$

0000000107ff9f90: 0a09 6c6f 6c2e 6c6f 6c0a 2000 ..lol.lol. .$

$>
```

• Նախատիպը պետք է լինի այսպիսին`

```
void *ft_print_memory(void *addr, unsigned int size);
```

• Պետք է վերադարձնի addr.

#### Գլուխ XVI

# <անձնում և ընկերն ընկերոջը ստուգում

Հանձնեք ձեր առաջադրանքը Git պահոցում, ինչպես սովորաբար անում եք։ Ստուգման ժամանակ գնահատվելու է միայն ձեր պահոցի պարունակությունը։ Մի՛ վարանեք նորից ստուգել ձեր ֆայլերի անունները՝ համոզվելու համար, որ դրանք ճիշտ են։



<արկավոր է հանձնել միայն այն ֆայլերը, որոնք պահանջվում են այս նախագիծը նկարագրող ֆայլում։