

C Ավազան C 11

<ակիրճ. այս փաստաթուղթը 42 դպրոցի C ավազանի C 11 մոդուլի նյութն t

Ցանկ

1	Յուցուսներ	2
II	Նախաբան	4
Ш	Առաջադրանը 00 ։ ft_foreach	5
IV	Առաջադրանք 01 ։ ft_map	6
\mathbf{V}	Առաջադրանք 02 ։ ft_any	7
VI	Առաջադրանը 03 ։ ft_count_if	8
VII	Առաջադրանը 04 ։ ft_is_sort	9
VIII	Առաջադրանք 05 ։ do-op	10
IX	Առաջադրանը 06 ։ ft_sort_string_tab	13
X	Առաջադրանը 07 ։ ft_advanced_sort_string_tab	14
XI	Հանձնում և ընկերն ընկերոջը ստուգում	15

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։

- 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։

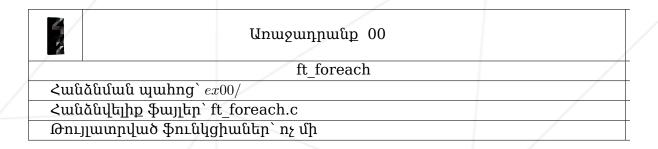
Գլուխ II Նախաբան

«Ես էլ քեզ եմ սիրում։ Անիմաստ է դրժել, որ քո հանդեպ ունեմ այնպիսի զգացմունքներ, որ որևէ այլ կերպ չեն կարող բացատրվել։ Մի կարճ ժամանակ ծանրութեթև էի անում ուղեղումս մակաբույծ ունենալու հավանականությունը, բայց դա էլ ավելի դժվարահավատ տարբերակ է։ Միակ տրամաբանական եզրակացությունը սերն է»։

Շելդոն Քուփեր, «Մեծ պայթյունի տեսությունը» հեռուստասերիալ

Գլուխ III

Uրաջադրանը 00 : ft_foreach



- Ստեղծել ft_foreach ֆունկցիան, որը կիրառում է ֆունկցիա տրված զանգվածի յուրաքանչյուր տարրի վրա։ Ֆունկցիան կկիրառվի ըստ զանգվածի տարրերի հերթականության։
- Նախատիպը պետք է լինի այսպիսին`

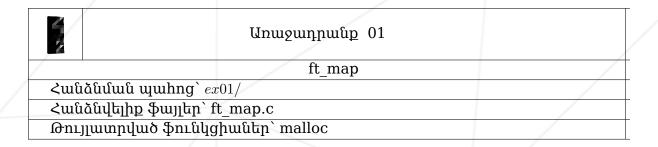
```
void ft_foreach(int *tab, int length, void(*f)(int));
```

• Օրինակ՝ ft_foreach ֆունկցիան կարող է կանչվել հետևյալ կերպ՝ զանգվածի բոլոր int-երր արտածելու համար։

ft_foreach(tab, 1337, &ft_putnbr);

Գլուխ IV

Առաջադրանք 01 : ft_map

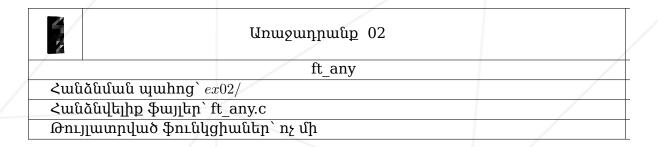


- Ստեղծել ft_map ֆունկցիան, որը կիրառում է ֆունկցիա տրված զանգվածի յուրաքանչյուր տարրի վրա (ըստ հերթականության) և վերադարձնում է բոլոր վերադարձվող արժեքների զանգվածը։
- Ֆունկցիան կկիրառվի ըստ զանգվածի տարրերի հերթականության։
- Նախատիպը պետք է լինի այսպիսին`

int *ft_map(int *tab, int length, int(*f)(int));

Գլուխ V

Առաջադրանը 02 : ft_any



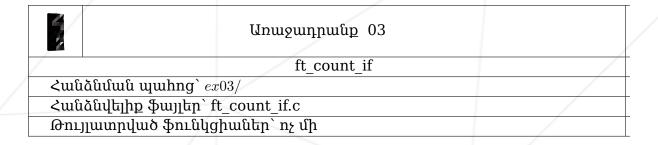
- Ստեղծել ft_any ֆունցիա, որը վերադարձնում է 1, եթե որպես պարամետր ստացած զանգվածի գոնե մեկ տարրի վրա f ֆունկցիայի կիրառության արդյունքը է 0 չէ։ Հակառակ դեպքում ֆունկցիան պետք է վերադարձնի 0։
- Ֆունկցիան կկիրառվի ըստ զանգվածի տարրերի հերթականության։
- Նախատիպը պետք է լինի այսպիսին`

int ft_any(char **tab, int(*f)(char*));

• Չանգվածը պետք է սահմանափակվի ոսև ցուցիչով։

Գլուխ VI

Uրաջադրանը 03 : ft_count_if

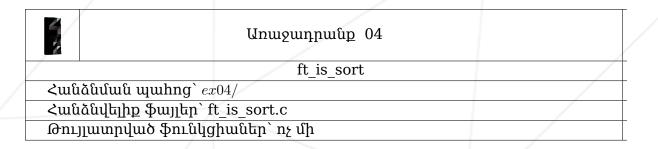


- Ստեղծել ft_count_if ֆունկցիան, որը կվերադարձնի զանգվածի այն տարրերի քանակը, որոնց վրա f ֆունցիայի կիրառության արդյունքը 0 չէ։
- Ֆունկցիան կկիրառվի ըստ զանգվածի տարրերի հերթականության։
- Նախատիպը պետք է լինի այսպիսին`

int ft_count_if(char **tab, int length, int(*f)(char*));

Գլուխ VII

Առաջադրանը 04 : ft_is_sort

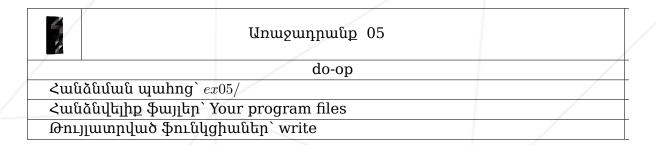


- Ստեղծել ft_is_sort ֆունցիան, որը վերադարձնում է 1, եթե զանգվածը դասավորված է և 0` հակառակ դեպքում։
- Որպես արգումենտ փոխանցված ֆունցիան պետք է վերադարձնի բացասական ամբողջ թիվ, եթե առաջին արգումենտը երկրորդից փոքր է, 0, եթե հավասար է երկրորդին, և դրական ամբողջ թիվ բոլոր այլ դեպքերում։
- Նախատիպը պետք է լինի այսպիսին`

nt ft_is_sort(int *tab, int length, int(*f)(int, int));

Գլուխ VIII

Առաջադրանք 05 : do-op



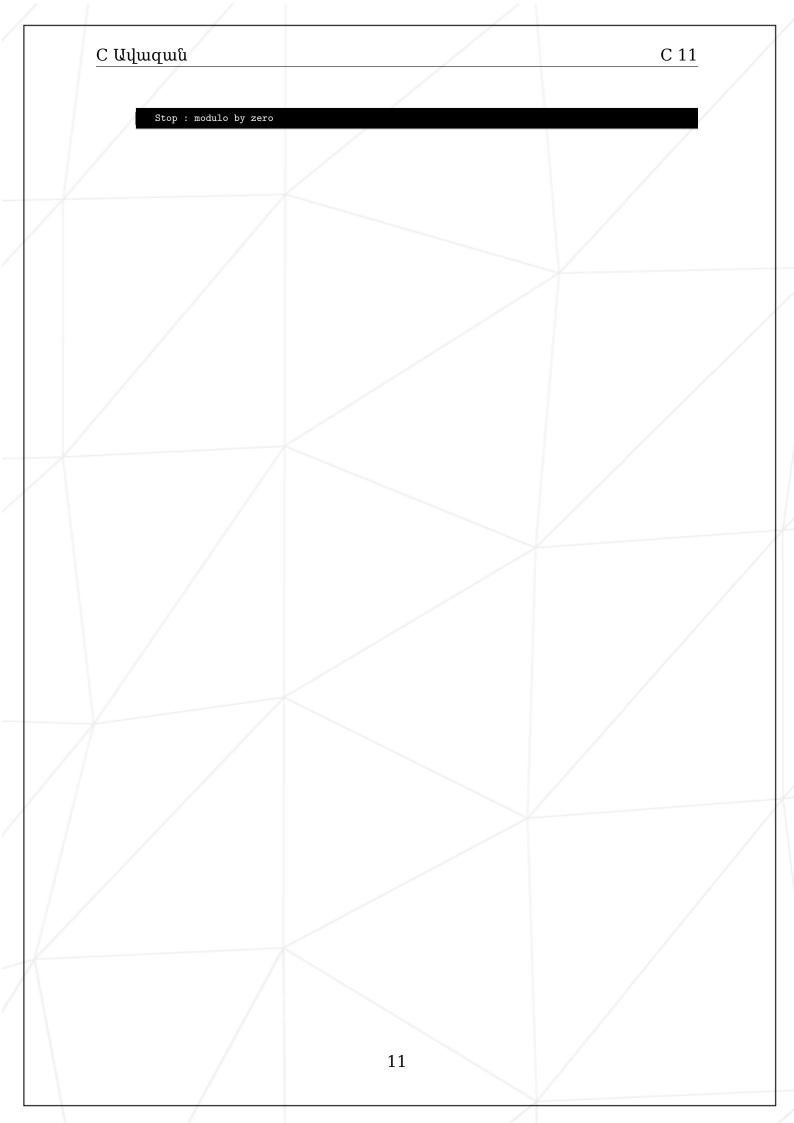
- Ստեղծել do-op անունով ծրագիր։
- Ծրագիրը կիրականացվի երեք արգումենաներով՝ do-op արժեք1 օպերատոր արժեք 2։
- Օրինակ՝

```
$>./do-op 42 "+" 21
63
$>
```

- Գործողությունը մշակելու համար անհրաժեշտ է օգտագործել ֆունկցիաների ցուցիչների զանգված։
- Անվավեր օպերատորի դեպքում ծրագիրը պետք է տպի 0։
- Եթե արգումենտների քանակը անվավեր է, do-op-ը ոչինչ չի արտածում։
- Ծրագիրը պետք է ընդունի և տպի արդյունքը հետևյալ օպերատորների համար` '+' '-' '/' '*' և '%':
- Ծրագիրը պետք է արժեքի հետ վարվի որպես int-ի։
- 0-ի վրա բաժանման դեպքում այն պետք է տպի՝

```
Stop : division by zero
```

• 0-ի վրա մնացորդով բաժանման կիրառման դեպքում այն պետք է տպի`



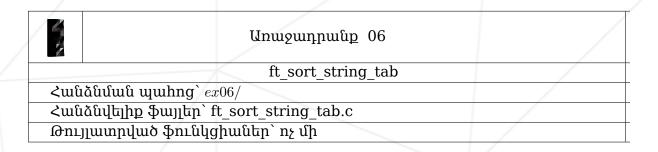
C Ավազան C 11

• Ահա Moulinette-ի կատարած թեստերի օրինակներ՝

```
$> make clean
$> make
$> ./do-op
$> ./do-op 1 + 1
2
$> ./do-op 42amis - --+-20toto12
62
$> ./do-op 1 p 1
0
$> ./do-op 1 + toto3
1
$>
$> ./do-op toto3 + 4
4
$$ ./do-op foo plus bar
0
$> ./do-op 25 / 0
Stop : division by zero
$> ./do-op 25 % 0
Stop : modulo by zero
$>
```

Գլուխ IX

Առաջադրանք 06 ։ ft_sort_string_tab

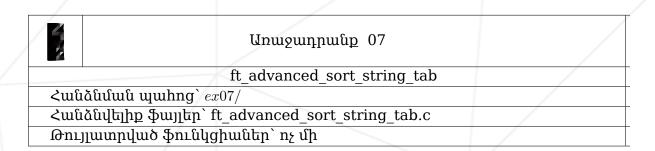


- Ստեղծել ft_sort_string_tab ֆունկցիան, որը դասավորում է tab-ի տողերը ascii հերթականությամբ։
- tab-ն ավարտվում է null-nվ։
- Դասավորումը կիրականացվի զանգվածի ցուցիչների փոխանակմամբ։
- Նախատիպը պետք է լինի այսպիսին`

void ft_sort_string_tab(char **tab);

Գլուխ X

Առաջադրանք 07 : ft_advanced_sort_string_tab



- Ստեղծել ft_advanced_sort_string_tab ֆունկցիան, որը դասավորում է` կախված որպես արգումենտ տրված ֆունկցիայի վերադարձրած արժեքից։
- Դասավորումը կիրականացվի զանգվածի ցուցիչների փոխանակմամբ։
- tab -ը ավարտվում է null արժեքով։
- Նախատիպը պետք է լինի այսպիսին։

void ft_advanced_sort_string_tab(char **tab, int(*cmp)(char *, char *));



Եթե ft_advanced_sort_string_tab() ֆունկցիան կանչենք` նրան որպես երկրորդ արգումենտ փոխանցելով ft_strcmp-ը, ապա վերադարձված արժեքը նույնը կլինի ինչ ft_sort_string_tab()-ինը։

Գլուխ XI

<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։



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