



C Ավագան

C 02

Հակիրճ. այս փաստաթուղթը 42 դպրոցի C ավագանի C 02 մոդուլի նյութն է:

# Ցանկ

<b>I</b>	<b>Ցուցումներ</b>	<b>2</b>
<b>II</b>	<b>Նախաբան</b>	<b>4</b>
<b>III</b>	<b>Առաջադրանք 00 : ft_strcpy</b>	<b>5</b>
<b>IV</b>	<b>Առաջադրանք 01 : ft_strncpy</b>	<b>6</b>
<b>V</b>	<b>Առաջադրանք 02 : ft_str_is_alpha</b>	<b>7</b>
<b>VI</b>	<b>Առաջադրանք 03 : ft_str_is_numeric</b>	<b>8</b>
<b>VII</b>	<b>Առաջադրանք 04 : ft_str_is_lowercase</b>	<b>9</b>
<b>VIII</b>	<b>Առաջադրանք 05 : ft_str_is_uppercase</b>	<b>10</b>
<b>IX</b>	<b>Առաջադրանք 06 : ft_str_is_printable</b>	<b>11</b>
<b>X</b>	<b>Առաջադրանք 07 : ft_strupcase</b>	<b>12</b>
<b>XI</b>	<b>Առաջադրանք 08 : ft_strlowcase</b>	<b>13</b>
<b>XII</b>	<b>Առաջադրանք 09 : ft_strcapitalize</b>	<b>14</b>
<b>XIII</b>	<b>Առաջադրանք 10 : ft_strlcpy</b>	<b>15</b>
<b>XIV</b>	<b>Առաջադրանք 11 : ft_putstr_non_printable</b>	<b>16</b>
<b>XV</b>	<b>Առաջադրանք 12 : ft_print_memory</b>	<b>17</b>

# Գլուխ I

## Ցուցումներ

- Այս էջը ձեր միակ ուղեցույցն է: Պատվող 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:
- Բացի դրանից, ձեր առաջադրանքները կստուգվեն և կգնահատվեն Moulinette կոչվող ծրագրով:
- Moulinette-ը գնահատելիս շատ բժախնդիր է ու խիստ: Այն ամբողջովին ավտոմատացված է, և գնահատման հարցում նրա հետ անհնար է բանակցել: Այսպիսով, տիպիկ անակնկալներից խուսափելու համար առաջադրանքները պետք է կատարվեն հնարավորինս անթերի:
- Moulinette-ն այնքան էլ լայնախոհ չէ: Այն չի էլ փորձի հասկանալ ձեր կողը, 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 gcc:
- Եթե ձեր ծրագիրը չկազմարկվի, կստանաք 0:
- Նյութում նշված ֆայլից բացի ձեր պահոցում հավելյալ 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» հատվածից 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 «Միլիկոնյան հովիտ» սերիալից:

- Լսի՛ր, ինչու՞ ուղղակի չգործածել Vim-ը Emacs-ի փոխարեն (ԾԻԾԱՂ):
- Ես հենց այդպես էլ անում եմ՝ Vim-ը Emacs-ի փոխարեն:
- Օ՛, 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:
- Ռիչարդ: (Անընդհատ սեղմում է բացատի կոճակը):
- Ա՛յ քեզ բան, լավ, հաջողություն:
- Մի tab-ը ուր բացատ է խնայում: (ԴՈՒՌԸ ՓԱԿՎՈՒՄ Է) - (ՇՐԸ՛ԽԿ)


...  
(Ռիչարդը տնքտնքում է)

- Աստվա՛ծ իմ, Ռի՛չարդ, ի՞նչ եղավ:
- Ուղղակի փորձեցի միանգամից ուր աստիճան իջնել, բայց ես լավ եմ, մի՛ անհանգստացիր:
- Ցտեսություն, Ռիչա՛րդ:
- Չուտ տեսակետս էի պաշտպանում:

Հուսով եմք՝ այս առաջադրանքը կատարելիս ձեզ չեն ստիպի emacs ու բացատի կոճակ գործածել:

## Գլուխ III

### Առաջադրանք 00 : ft\_strcpy


	Առաջադրանք 00
	ft_strcpy
	Հանձնման պահոց՝ ex00/
	Հանձնվելիք ֆայլեր՝ ft_strcpy.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի

- Վերարտադրել strcpy (man strcpy) ֆունկցիայի վարքը:
- Նախատիպը պետք է լինի այսպիսին՝

```
char *ft_strcpy(char *dest, char *src);
```

## Գլուխ IV

### Առաջադրանք 01 : ft\_strncpy


	Առաջադրանք 01
	ft_strncpy
	Հանձնման պահոց՝ ex01/
	Հանձնվելիք ֆայլեր՝ ft_strncpy.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի

- Վերարտադրել strncpy (man strncpy) ֆունկցիայի վարքը:
- Նախատիպը պետք է լինի այսպիսին՝

```
char *ft_strncpy(char *dest, char *src, unsigned int n);
```

## Գլուխ V

### Առաջադրանք 02 : ft\_str\_is\_alpha

	Առաջադրանք 02
	ft_str_is_alpha
	Հանձնման պահոց՝ ex02/
	Հանձնվելիք ֆայլեր՝ ft_str_is_alpha.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի

- Գրել ֆունկցիա, որը վերադարձնում է 1, եթե որպես պարամետր տրված 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`

	Առաջադրանք 03
	<code>ft_str_is_numeric</code>
	Հանձնման պահոց՝ <i>ex03/</i>
	Հանձնվելիք ֆայլեր՝ <code>ft_str_is_numeric.c</code>
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- Գրել ֆունկցիա, որը վերադարձնում է 1, եթե որպես պարամետր տրված 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

	Առաջադրանք 04
	ft_str_is_lowercase
	Հանձնման պահոց՝ ex04/
	Հանձնվելիք ֆայլեր՝ ft_str_is_lowercase.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- Գրել ֆունկցիա որը վերադարձնում է 1, եթե որպես պարամետր տրված 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

	Առաջադրանք 05
	ft_str_is_uppercase
	Հանձնման պահոց՝ ex05/
	Հանձնվելիք ֆայլեր՝ ft_str_is_uppercase.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- 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);
```

- Պետք է վերադարձնի 1, եթե str դատարկ է:

## Գլուխ IX

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

	Առաջադրանք 06
	ft_str_is_printable
	Հանձնման պահոց՝ ex06/
	Հանձնվելիք ֆայլեր՝ ft_str_is_printable.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- Գրել ֆունկցիա որը վերադարձնում է 1, եթե որպես պարամետր տրված 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

	Առաջադրանք 07
ft_strupcase	
Հանձնման պահոց՝ ex07/	
Հանձնվելիք ֆայլեր՝ ft_strupcase.c	
Թույլատրված ֆունկցիաներ՝ ոչ մի	


- 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

	Առաջադրանք 08
	ft_strlowcase
	Հանձնման պահոց՝ ex08/
	Հանձնվելիք ֆայլեր՝ ft_strlowcase.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- Գրել ֆունկցիա, որը բոլոր տառերը դարձնում է փոքրատառ:
- Նախատիպը պետք է լինի այսպիսին՝

```
char *ft_strlowcase(char *str);
```

- Պետք է վերադարձնի str.

## Գլուխ XII

### Առաջադրանք 09 : ft\_strcapitalize

	Առաջադրանք 09
	ft_strcapitalize
	Հանձնման պահոց՝ ex09/
	Հանձնվելիք ֆայլեր՝ ft_strcapitalize.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի

- 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

### Առաջադրանք 10 : ft\_strlcpy

	Առաջադրանք 10
	ft_strlcpy
	Հանձնման պահոց՝ ex10/
	Հանձնվելիք ֆայլեր՝ ft_strlcpy.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի


- 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

	Առաջադրանք 11
	ft_putstr_with_non_printable
	Հանձնման պահոց՝ ex11/
	Հանձնվելիք ֆայլեր՝ ft_putstr_non_printable.c
	Թույլատրված ֆունկցիաներ՝ write

- 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\0atu vas bien ?
```

- Նախատիպը պետք է լինի այսպիսին՝

```
void ft_putstr_non_printable(char *str);
```

## Գլուխ XV

### Առաջադրանք 12 : ft\_print\_memory

	Առաջադրանք 12
	ft_print_memory
	Հանձնման պահոց՝ ex12/
	Հանձնվելիք ֆայլեր՝ ft_print_memory.c
	Թույլատրված ֆունկցիաներ՝ write

- Գրել ֆունկցիա, որը հիշողության տիրույթը արտապատկերում է էկրանին:

Հիշողության տիրույթի արտապատկերումը պետք է բաժանված լինի երեք սյուների, որոնք էլ իրարից պետք է առանձնացված լինեն բացատով:

- Տասնվեցական համակարգով տրված հասցեի առաջին տողի առաջին նիշը , 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):
- Այդ հատվածը՝ ներկայացված է տավոդ նիշերով:
- Եթե նիշը չարտածվող է, ապա այն կփոխարինվի կետով:
- Յուրաքանչյուր տող պետք է պարունակի տասնվեց նիշ:
- Եթե չափ/երկարությունը 0 է, ոչինչ չի արտապատկերվի:

- Օրինակ՝

```
$> ./ft_print_memory
000000010a161f40: 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.