



Շ Ավագան

Շ 05

Հակիրճ. այս փաստաթուղթը 42 դպրոցի Շ ավագանի Շ 05 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_iterative_factorial</b>	<b>6</b>
<b>IV</b>	<b>Առաջադրանք 01 : ft_recursive_factorial</b>	<b>7</b>
<b>V</b>	<b>Առաջադրանք 02 : ft_iterative_power</b>	<b>8</b>
<b>VI</b>	<b>Առաջադրանք 03 : ft_recursive_power</b>	<b>9</b>
<b>VII</b>	<b>Առաջադրանք 04 : ft_fibonacci</b>	<b>10</b>
<b>VIII</b>	<b>Առաջադրանք 05 : ft_sqrt</b>	<b>11</b>
<b>IX</b>	<b>Առաջադրանք 06 : ft_is_prime</b>	<b>12</b>
<b>X</b>	<b>Առաջադրանք 07 : ft_find_next_prime</b>	<b>13</b>
<b>XI</b>	<b>Առաջադրանք 08 : Տասը թագուհիները</b>	<b>14</b>
<b>XII</b>	<b>Հանձնում և ընկերն ընկերոջը ստուգում</b>	<b>16</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 բծախնդիր 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 `cc`:
- 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

Հավանաբար, հետևյալ կատակը կօգնի ավելի հեշտ կատարել առաջադրանքները:

Ուստիկանը հարցաքննում է կասկածյալին:

ՈՍՏԻԿԱՆ.- Պարո՛ն Պետրոսյան, որտեղի՞ց Ձեզ այդքան գումար:

ՊԵՏՐՈՍՅԱՆ.- Մեր տան դարակից:

ՈՍՏԻԿԱՆ.- Իսկ ո՞վ է գումարը դրել դարակի մեջ:

ՊԵՏՐՈՍՅԱՆ.- Կինս:

ՈՍՏԻԿԱՆ.- Իսկ Ձեր կնո՞ջն ով է տվել այդ գումարը:

ՊԵՏՐՈՍՅԱՆ.- Ես եմ տվել:

ՈՍՏԻԿԱՆ.- Իսկ որտեղի՞ց Ձեզ այդքան գումար:

ՊԵՏՐՈՍՅԱՆ.- Մեր տան դարակից:

ՈՍՏԻԿԱՆ.- Իսկ ո՞վ է գումարը դրել դարակի մեջ:

ՊԵՏՐՈՍՅԱՆ.- Կինս:

ՈՍՏԻԿԱՆ.- Իսկ Ձեր կնո՞ջն ով է տվել այդ գումարը:

ՊԵՏՐՈՍՅԱՆ.- Ես եմ տվել:

ՈՍՏԻԿԱՆ.- Իսկ որտեղի՞ց Ձեզ այդքան գումար:


.....

Հիմա ես գիտեմ, որ դու արդեն գիտես, թե ինչ բան է ռեկուրսիան: Ես

նաև գիտեմ, որ դու գիտես, որ ես գիտեմ, որ դու գիտես, որ ես գիտեմ,  
որ դու գիտես, ..., թե ինչ է ռեկուրսիան:

## Գլուխ III

# Առաջադրանք 00 : ft\_iterative\_factorial


	Առաջադրանք 00
	ft_iterative_factorial
	Հանձնման պահոց՝ <i>ex00/</i>
	Հանձնվելիք ֆայլեր՝ ft_iterative_factorial.c
	Թույլատրված ֆունկցիաներ՝ ոչ մի

- Ստեղծել իտերացիոն ֆունկցիա, որը վերադարձնում է թիվ: Այդ թիվը որպես մուտքային պարամետր փոխանցված թվի ֆակտորիալն է:
- Եթե արգումենտը 0-ից փոքր է, ֆունկցիան վերադարձնում է 0:
- Գերհագեցումները չպետք է մշակվեն, ֆունկցիայի վերադարձը կլինի անորոշ:
- 0-ի 0 աստիճանը պետք է 1 վերադարձնի:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_iterative_factorial(int nb);
```

## Գլուխ IV

### Առաջադրանք 01 : ft\_recursive\_factorial

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


- Ստեղծել ռեկուրսիվ ֆունկցիա, որը վերադարձնում է որպես մուտքային պարամետր փոխանցված թվի ֆակտորիալը:
- Եթե արգումենտը վավեր չէ, ֆունկցիան վերադարձնում է 0:
- Գերհագեցումները չպետք է մշակվեն, ֆունկցիայի վերադարձը կլինի անորոշ:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_recursive_factorial(int nb);
```



## Գլուխ V

### Առաջադրանք 02 : ft\_iterative\_power


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

- Ստեղծել իտերացիոն ֆունկցիա, որը վերադարձնում է աստիճան բարձրացրած թվի արժեքը: 0-ից փոքր աստիճանի դեպքում վերադարձվում է 0: Գերհագեցումները չպետք է մշակվեն:
- Համարվում է, որ 0-ի 0 աստիճանը 1 է:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_iterative_power(int nb, int power);
```

## Գլուխ VI

### Առաջադրանք 03 : ft\_recursive\_power


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

- Ստեղծել ռեկուրսիվ ֆունկցիա, որը վերադարձնում է աստիճան բարձրացրած թվի արժեքը:
- Գերհագեցումները չպետք է մշակվեն, ֆունկցիայի վերադարձը կլինի անորոշ:
- Համարվում է, որ 0-ի 0 աստիճանը 1 է:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_recursive_power(int nb, int power);
```

## Գլուխ VII

### Առաջադրանք 04 : ft\_fibonacci

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


- Ստեղծել ft\_fibonacci ֆունկցիա, որը կվերադարձնի Ֆիբոնաչիի հաջորդականության n-րդ տարրը: Առաջին տարրը 0 ինդեքսում է: Կհամարենք, որ Ֆիբոնաչիի հաջորդականությունը սկսվում է այսպես՝ 0, 1, 1, 2:
- Գերհագեցումները չպետք է մշակվեն, ֆունկցիայի վերադարձը կլինի անորոշ:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_fibonacci(int index);
```

- Բնականաբար, ft\_fibonacci-ն պետք է լինի ռեկուրսիվ:
- Եթե index-ը 0-ից փոքր է, ֆունկցիան պետք է վերադարձնի -1:

## Գլուխ VIII

### Առաջադրանք 05 : ft\_sqrt


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

- Ստեղծել ֆունկցիա, որը վերադարձնում է թվի քառակուսի արմատը (եթե այն գոյություն ունի), կամ 0, եթե թվի արմատն իրացիոնալ թիվ է:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_sqrt(int nb);
```

## Գլուխ IX

### Առաջադրանք 06 : ft\_is\_prime

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

- Ստեղծել ֆունկցիա, որը վերադարձնում է 1, եթե որպես պարամետր փոխանցված թիվը պարզ թիվ է և 0, եթե պարզ չէ:
- Նախատիպը պետք է լինի այսպիսին՝


```
int ft_is_prime(int nb);
```



0-ն և 1-ը պարզ թվեր չեն:

## Գլուխ X

### Առաջադրանք 07 : `ft_find_next_prime`


	Առաջադրանք 07
<code>ft_find_next_prime</code>	
Հանձնման պահոց՝ <i>ex07/</i>	
Հանձնվելիք ֆայլեր՝ <code>ft_find_next_prime.c</code>	
Թույլատրված ֆունկցիաներ՝ ոչ մի	

- Ստեղծել ֆունկցիա, որը վերադարձնում է որպես արգումենտ փոխանցված թվից մեծ կամ հավասար հաջորդ պարզ թիվը:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_find_next_prime(int nb);
```

## Գլուխ XI

# Առաջադրանք 08 : Տասը թագուհիները

	Առաջադրանք 08
Տասը թագուհիները	
Հանձնման պահոց՝ <i>ex08/</i>	
Հանձնվելիք ֆայլեր՝ <i>ft_ten_queens_puzzle.c</i>	
Թույլատրված ֆունկցիաներ՝ <i>write</i>	

- Ստեղծել ֆունկցիա, որն արտածում է տասը սյունակ և տասը տող ունեցող շախմատի տախտակի վրա տասը թագուհիների բոլոր հնարավոր դիրքերը՝ առանց թույլ տալու թագուհիներին իրար հարվածել մեկ քայլով, և վերադարձնում է հնարավոր դիրքերի քանակը:
- Այս խնդիրը լուծելու համար պետք է օգտագործել ռեկուրսիա:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_ten_queens_puzzle(void);
```

- Այն պետք է արտապատկերվի հետևյալ կերպ՝

```
$>./a.out | cat -e
0257948136$
0258693147$
...
4605713829$
4609582731$
...
9742051863$
$>
```

- Հաջորդականությունը ձախից աջ է: Առաջին թիվը ցույց է տալիս առաջին թագուհու դիրքն առաջին սյունակում (ինդեքսավորումը

սկսվում է 0-ից): N-րդ թիվը ցույց է տալիս N-րդ թագուհու դիրքը N-րդ սյունակում:

- Վերադարձվող արժեքը պետք է լինի արտածված լուծումների ընդհանուր քանակը:



## Գլուխ XII

# Հանձնում և ընկերն ընկերոջը ստուգում

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



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