



Շ Ավագան

Շ 07

Հակիրճ. այս փաստաթուղթը 42 դպրոցի Շ ավագանի Շ 07 մոդուլի
նյութն է:

Ցանկ

I	Ցուցումներ	2
II	Նախաբան	4
III	Առաջադրանք 00 : ft_strdup	6
IV	Առաջադրանք 01 : ft_range	7
V	Առաջադրանք 02 : ft_ultimate_range	8
VI	Առաջադրանք 03 : ft_strjoin	9
VII	Առաջադրանք 04 : ft_convert_base	10
VIII	Առաջադրանք 05 : ft_split	11
IX	Հանձնում և ընկերն ընկերոջը ստուգում	12

Գլուխ I

Ցուցումներ

- 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:
- 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 `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

Ահա մի հատված «Հարրի Փոթերը» վիպաշարից (թարգմանիչ՝ Լիլիթ Մկրտչյան):

Գուցե մտածես՝ գեղեցիկ չեմ ես,
Եվ կսխալվես, թե տեսածով դատես:
Երդվում եմ կոտրած ճարմանդովս միակ,
Որ չկա աշխարհում ինձ նման գլխարկ:

Բայց թե կասկածես ու շատ փնտրտես,
Քիթս կկտրեմ, թե ուրիշը գտնես:
Դե՛ն նետեք բերետ, փափախ ու կեպի՝
Գլխարկներ դատարկ, ապուշ, անհոգի:

ԵՍ եմ այստեղ միայն կարևոր, ամենագետ ու գլխավոր,
Գլխարկն եմ ես անգուգական, իմաստուն ու գորավոր:
Տեսակավորող Գլխարկն եմ ես՝ գլխարկների արքան հեռատես.
Թե վստահես ու գլխիդ ինձ դնես, ճակատագիրդ այստեղ կտեսնես:

Դու երբեք ինձնից ոչինչ չես թաքցնի,
Դի՛ր գլխիդ վստահ ու մի՛ վարանի:
Կասեմ քեզ շիտակ, թե ուր կգնաս,
Ինչ գործեր կանես, ում կընկերանաս:

Գուցե սպասում են քեզ Գրիֆինդորո՞ւմ,
Ուր աճել են բազում դյուրթագներ քաջանուն,
Նրանց սիրտը խիզախ է, ոգին՝ անկոտրում,
Փա՛նք ու պատի՛վ են Գրիֆինդոր բերում:

Թե արդարամիտ ես ու համբերատար,
Գործիդ նվիրված ու կարգապահ,
Գուցե մտնես դու Հաֆլիփա՞ֆ ընտանիք,
Փա՛նք Ձեզ, արդար հաֆլիփաֆցիք:

Գուցե ընտրվես դու Ռեյվենքլո՞ Տուն,
Թե խելքդ է ուսյալ, և միտքդ՝ գիտուն,

Գրքի բարեկամ ես ու իրավ ջանասեր,
Փա՛ռք Ռեյվենքլո Տանն իմաստասեր:


Անշուշտ կգտնես ընկերներ բազում,
Թե՛ չես պարտվում երբեք ու չես նահանջում,
Կընտրվես ճարպիկ, խորամանկ Սլիգերին,
Փա՛ռք սլիգերինցի հնարամիտներին:

Դի՛ր ինձ գլխիդ ու մի՛ տատանվիր,
Թո՛ղ ճշմարիտ ընտրություն կատարվի,
Ինձնից լավ ու ազնիվ բարեկամ դու չունես,
Տեսակավորող Գլխարկն եմ ես:

Յավոք, այս նյութը որևէ կապ չունի Հարրի Փոթթերի հետ: Ավելի վատ
ձեզ համար, որովհետև հույսների չեք կարող դնել *կախարդանքի* վրա:

Գլուխ III

Առաջադրանք 00 : ft_strdup


	Առաջադրանք 00
ft_strdup	
Հանձնման պահոց՝ ex00/	
Հանձնվելիք ֆայլեր՝ ft_strdup.c	
Թույլատրված ֆունկցիաներ՝ malloc	

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

```
char *ft_strdup(char *src);
```

Գլուխ IV

Առաջադրանք 01 : ft_range

	Առաջադրանք 01
	ft_range
	Հանձնման պահոց՝ ex01/
	Հանձնվելիք ֆայլեր՝ ft_range.c
	Թույլատրված ֆունկցիաներ՝ malloc


- Ստեղծել ft_range ֆունկցիա, որը վերադարձնում է int-երի զանգված: Այս int զանգվածը պետք է պարունակի min և max միջակայքի բոլոր արժեքները:
- Min-ը ներառյալ, max-ը՝ ոչ:
- Նախատիպը պետք է լինի այսպիսին՝

```
int *ft_range(int min, int max);
```

- Եթե min-ի արժեքը մեծ է կամ հավասար max-ի արժեքին, null ցուցիչ պետք է վերադարձվի:

Գլուխ V

Առաջադրանք 02 : ft_ultimate_range

	Առաջադրանք 02
	ft_ultimate_range
	Հանձնման պահոց՝ ex02/
	Հանձնվելիք ֆայլեր՝ ft_ultimate_range.c
	Թույլատրված ֆունկցիաներ՝ malloc


- Ստեղծել ֆունկցիա, որը ստեղծում և վերագրում է int-երի զանգված: int զանգվածը պետք է պարունակի min և max միջակայքի բոլոր արժեքները:
- Min-ը ներառյալ, max՝ ոչ:
- Նախատիպը պետք է լինի այսպիսին՝

```
int ft_ultimate_range(int **range, int min, int max);
```

- range-ի չափը պետք է վերադարձվի (կամ -1 սխալի դեպքում):
- Եթե min-ի արժեքը մեծ կամ հավասար է max-ի արժեքին, range-ը ցույց կտա NULL արժեքի վրա, և կվերադարձվի 0:

Գլուխ VI

Առաջադրանք 03 : ft_strjoin


	Առաջադրանք 03
	ft_strjoin
	Հանձնման պահոց՝ ex03/
	Հանձնվելիք ֆայլեր՝ ft_strjoin.c
	Թույլատրված ֆունկցիաներ՝ malloc

- Ստեղծել ֆունկցիա, որը կցում է բոլոր տողերը, որոնց վրա ցույց է տալիս `strs`-ը՝ բաժանված `sep`-ով: `size`-ը `strs`-ում տողերի քանակն է:
- Եթե `size`-ը 0 է, պետք է վերադարձվի դատարկ տող, որը կարող եք ազատել `free()`-ով:
- Նախատիպը պետք է լինի այսպիսին՝

```
char *ft_strjoin(int size, char **strs, char *sep);
```

Գլուխ VII

Առաջադրանք 04 : ft_convert_base


	Առաջադրանք 04
ft_convert_base	
Հանձնման պահոց՝ ex04/	
Հանձնվելիք ֆայլեր՝ ft_convert_base.c, ft_convert_base2.c	
Թույլատրված ֆունկցիաներ՝ malloc, free	

- Ստեղծել ֆունկցիա, որը վերադարձնում է տողի՝ base_from համակարգից base_to համակարգ փոխակերպման արդյունքը:
- nbr, base_from, base_to-ն կարող են տպելի չլինել: nbr-ն պետք է ենթարկվի նույն կանոններին ինչ ft_atoi_base-ը (մյուս մոդուլից): Չգուշացեք «+», «-» և բացատի նշաններից:
- nbr-ով ներկայացված թիվը պետք է տեղավորվի int-ում:
- Միայն համակարգի դեպքում պետք է վերադարձվի NULL:
- Վերադարձված թիվը կարող է նախորդվել միայն մեկ '-' նշանով՝ անհրաժեշտության դեպքում: Ոչ մի '+' նշան կամ սպիտակ տարածք:
- Նախատիպը պետք է լինի այսպիսին՝

```
char *ft_convert_base(char *nbr, char *base_from, char *base_to);
```

Գլուխ VIII

Առաջադրանք 05 : ft_split

	Առաջադրանք 05
	ft_split
	Հանձնման պահոց՝ ex05/
	Հանձնվելիք ֆայլեր՝ ft_split.c
	Թույլատրված ֆունկցիաներ՝ malloc

- Ստեղծել ֆունկցիա, որը բաժանում է տողի նիշերը՝ կախված այլ տողի նիշերից:
- Անհրաժեշտ է օգտագործել charset տողի յուրաքանչյուր նիշը որպես բաժանիչ:
- Ֆունկցիան պետք է վերադարձնի մի զանգված, որի ամեն տարրը պարունակում է երկու բաժանիչների միջև տողի հասցեն: Չանգվածի վերջին տարրը պետք է հավասար լինի 0-ի՝ ցույց տալու համար զանգվածի ավարտը:
- Չանգվածում չեն կարող լինել դատարկ տողեր: Հետևաբար, արե՛ք համապատասխան հետևությունները:
- Որպես արգումենտ տրված տողը պետք է լինի չփոփոխվող:
- Նախատիպը պետք է լինի այսպիսին՝

```
char **ft_split(char *str, char *charset);
```

Գլուխ IX

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: