

## Завдання на курсову роботу

1. Цільова мова транслятора асемблер (іх86).
2. Для отримання виконавчого файлу на виході розробленого транслятора скористатися програмами tasm.exe і tlink.exe або tasm32.exe і tlink32.exe.
3. мова розробки транслятора: ANSI C або C++.
4. Реалізувати оболонку або інтерфейс з командного рядка.
5. На вхід розробленого транслятора має подаватися текстовий файл, написаний на заданій мові програмування.
6. На виході розробленого транслятора мають створюватись чотири файли:  
*файл з повідомленнями про помилки (або про їх відсутність);*  
*файл на мові асемблера;*  
*об'єктний файл;*  
*виконавчий файл.*
7. Назва вхідної мови програмування утворюється від першої букви у прізвищі студента та останніх двох цифр номера його варіанту. Саме таке розширення повинні мати текстові файли, написані на цій мові програмування.

- Зауваження.**
- 1) В кожному завданні передбачається блок оголошення змінних;
  - 2) В кожному завданні обов'язковим є оператор присвоєння та оператор типу – блок;
  - 3) Оператори можуть бути довільної вкладеності і в любій послідовності.
  - 4) Для непарних варіантів завдань необхідно реалізувати висхідний метод граматичного розбору, а для парних варіантів завдань - нисхідний метод.

№ з/п	Блок тіла програми	Оператори уводу-виводу	Оператор присвоєння	Оператор	Регістр ключових слів	Регістр ідентифікаторів	Операції			Типи даних	Коментар	
							арифметичні	порівняння	логічні			
61	name <name>; body data...; end	READ WRITE	::	if - then goto; goto (Бейсик)	Up	Low2	+, -, *, /; %	=, <>, >, <;	!; and; or	INT16	{* ... *}	
62	NAME <name>; DATA...; BODY - END	Input Output	->	WHILE (CІ)	Up-Low перший символ Up	Up-Low4 перший символ Up	ADD; SUB; MUL; DIV; MOD	=, <>, >=, <=	!; &;	Int32	??...	
63	Program <name>; Begin Var...; End	scan print	<-	For (CІ)	Low	Low-Up4 перший символ Low	+, -; Mul; Div; Mod	==, !=, >, <	!; &&;	int_2	%% ... %%	
64	program <name>; var...; begin - end	GET PUT	>>	do- while (CІ)	Up	Up4	add; sub; *, /; %	==, !=, >=, <=	not; and; or	INT_4	\$\$ ...	
65	PROGRAM <name>; BEGIN VAR...; END	Read Write	::=	WHILE - DO (Паскаль)	Up-Low перший символ Up	Low4	+, -; *, DIV; MOD	==, !=, !>, !<	!; AND; OR	Int16_t	/* ... */	
66	Program <name>; Var...; Begin – End	input output	::=	For – DownTo (Паскаль)	Low	Up-Low6 перший символ Up	++, --; **, Div; Mod	=, <>, !>, !<	!; &;	int32_t	!! ...	
67	program <name>; begin var...; end	SCAN PRINT	<<	for –to - do(Паскаль)	Up	Low-Up6 перший символ Low	+, -; *, /; %	==, <>, >=, <=	!; &&;	INTEGER	// ...	
68	PROGRAM <name>; VAR...; BEGIN - END	Get Put	::	REPEAT – UNTIL (Паскаль)	Up-Low перший символ Up	Up6	ADD; SUB; MUL; DIV; MOD	=, !=, <, >	NOT; AND; OR	Integer16	// ... //	
69	Program <name>; Start Var ...; Finish	read write	->	For – To – Next (Бейсик)	Low	Low6	+, -; Mul; Div; Mod	Eq; Neq; Ls; Gr	!; And; Or	integer32	## ... ##	
70	program <name>; var ...; start - finish	INPUT OUTPUT	<-	if - then [- else] (Паскаль)	Up	Up-Low8 перший символ Up	add; sub; *, /; %	eq; noteq; less; gr	!; &;	INTEGER_2	\\ ...	

71	PROGRAM <name>; START VAR ...; FINISH	Scan Print	>>	IF [-ELSE] (CI)	Up-Low перший символ Up	Low-Up8 перший символ Low	+, -; *, DIV; MOD	EQ; NE; LT; GT	!; &&;	Integer_4	@ @ ...	
72	Program <name>; Var ...; Start - Finish	get put	:=	If - Then Goto; Goto (Бейсік)	Low	Up8	++, --; **, Div; Mod	Eg; Ne; >; <	Not; And; Or	integer16_t	## ...	
73	#program <name>; start variable...; stop	READ WRITE	::=	if - goto; goto (Бейсік)	Up	Low8	+, -; *, /; %	eg; ne; >>; <<	!; and; or	INTEGER32_t	{ ... }	
74	# PROGRAM <name>; VARIABLE ...; START – STOP	Input Output	<<	WHILE (CI)	Up-Low перший символ Up	Up-Low2 перший символ _	ADD; SUB; MUL; DIV; MOD	==; !=; LT; GT	!; &;	Longint	!! ...!!	
75	#Program <name>; Start Variable...; Stop	scan print	::	Do- While (CI)	Low	Low-Up2 перший символ _	+, -; Mul; Div; Mod	=; <>; Lt; Gt	!; &&;	int16	/* ...	
76	#program <name>; variable...; start – stop	GET PUT	->	for (CI)	Up	Up4 перший символ _	add; sub; *, /; %	eg; ne; le; ge	not; and; or	INT32	## ... ##	
77	# PROGRAM <name>; START VARIABLE ...; STOP	Read Write	<-	WHILE - DO (Паскаль)	Up-Low перший символ Up	Low4 перший символ _	+, -; *, DIV; MOD	EQ; NE; >; <=	!; AND; OR	Int_2	{* ... *}	
78	StartProgram Variable...; StartBlok -EndBlok	input output	>>	Repeat – Until (Паскаль)	Low	Up-Low4 перший символ _	++, --; **, Div; Mod	=; <>; Le; Ge	!; &;	int_4	??...	
79	startprogram startblok variable...; endblok	SCAN PRINT	:=	for –to - do(Паскаль)	Up	Low-Up4 перший символ _	+, -; *, /; %	==; !=; le; ge	!; &&;	INT16_t	% % ... % %	
80	STARTPROGRAM VARIABLE...; STARTBLOK - ENDBLOK	Get Put	::=	FOR – DOWNT0 (Паскаль)	Up-Low перший символ Up	Up6 перший символ _	ADD; SUB; MUL; DIV; MOD	=; <>; >; <;	NOT; AND; OR	Int32_t	\$\$ ...	
81	StartProgram StartBlok Variable...; EndBlok	read write	<<	For – To – Next (Бейсік)	Low	Low6 перший символ _	+, -; Mul; Div; Mod	=; <>; >; <=	!; And; Or	integer	/* ... */	
82	startprogram variable...; startblok - endblok	INPUT OUTPUT	::	if [-else] (CI)	Up	Up-Low6 перший символ _	add; sub; *, /; %	==; !=; >; <	!; &;	INTEGER16	!! ...	
83	MAIMPROGRAM START DATA...; END	Scan Print	->	IF - THEN [- ELSE] (Паскаль)	Up-Low перший символ Up	Low-Up6 перший символ _	+, -; *, DIV; MOD	==; !=; >; <=	NOT; AND; OR	Integer32	!! ...	
84	MainProgram Data...; Start - End	get put	<-	For (CI)	Low	Up8 перший символ _	++, --; **, Div; Mod	==; !=; !>; !<	!; &&;	integer32	// ...	
85	mainprogram start data...; end	READ WRITE	::	repeat – until (Паскаль)	Up	Low8 перший символ _	+, -; *, /; %	=; <>; !>; !<	!; &&;	INTEGER_2	// ... //	
86	MAIMPROGRAM DATA...; START - END	Input Output	->	FOR –TO – DO (Паскаль)	Up-Low перший символ Up	Low8 перший символ _	ADD; SUB; MUL; DIV; MOD	==; <>; >>; <<	NOT; AND; OR	Integer_4	## ... ##	
87	MainProgram Start Data...; End	scan print	>>	Do- While (CI)	Low	Up-Low8 перший символ _	+, -; Mul; Div; Mod	=, !=, <<, >>	!; And; Or	integer16_t	\\ ...	
88	name <name>; data...; body- end	GET PUT	:=	do- while (CI)	Up	Low-Up8 перший символ _	add; sub; *, /; %	eq; neq; ls; gr	!; &;	INTEGER32_t	{ ... }	
89	NAME <name>; BODY DATA...; END	Read Write	<<	FOR (CI)	Up-Low перший символ Up	Up-Low2 перший символ Up	+, -; *, DIV; MOD	EQ; NOTEQ; LESS; GR	!; &&;	Longint	!! ...!!	
90	Name <name>; Data...; Body -End	read write	::	While – Do (Паскаль)	Low	Low-Up2 перший символ Low	++, --; **, Div; Mod	Eg; Ne; Lt; Gt	Not; And; Or	int	/* ...	