



**Tshwane University
of Technology**

We empower people

**FORMAL SUMMATIVE ASSESSMENT
INSTRUCTIONS:**

1. All rules and regulations regarding student conduct and behaviour, as agreed upon by students at registration and as otherwise stated by the Tshwane University of Technology, apply.
2. A student card or proof of registration, along with an original identification document, must be presented to invigilators.
3. Ensure that your student number is correctly indicated on all submissions (paper-based or online) and that the attendance record is signed as required.
4. If a student is suspected of any form of cheating or plagiarism, either during the assessment or afterwards, the examiner, in their professional judgment and after consultation with the appointed moderator, may, at minimum, award a zero mark for the assessment. Further action may be taken against the student.

**DEPARTMENT OF
ELECTRICAL ENGINEERING**

**FACULTY OF
ENGINEERING AND THE BUILT
ENVIRONMENT**

**MODULE CODE
ES216AB**

**MODULE NAME
ENGINEERING SOFTWARE DESIGN A**

**ASSESSMENT NAME
Evaluation 1 - B**

EXAMINER:	Mr D. Engelbrecht Prof J.A. Jordaan	MARKS:	55 points
MODERATOR:	Mr A.J.Smith	PAGES:	11 (incl. cover)
DATE:	18 March 2025 13:00 to 14:30	TIME:	1 hour & 20minutes

STUDENT NUMBER:								
SURNAME:			INITIALS:			SIGNATURE:		

NOTE THE FOLLOWING:

1. **Plagiarism Policy:** Original work is required. We will use similarity detection software to review all student submissions for plagiarism. Ensure your work is your own.
2. **Internet Protocol (IP) Tracking:** IP addresses will be recorded and checked to verify that you have uploaded your work from the correct TUT laboratory.
3. **No External Devices:** The use of USB or other external devices is prohibited during the evaluation.
4. **Internet Access:** External internet access is not permitted.
5. **Evaluation Content:** This evaluation will cover topics from Unit 1 to Unit 3
6. **Programming Language:** Write your program in C, adhering to structured programming principles.
7. **Editing Requirements:** Your program must comply with all specified requirements. Refer to the appendices and attachments for more details.
8. **Submission Format:** Submit your source code file in the format
 <student number>.c , for example, 217123456.c
 (ONLY YOUR STUDENT NUMBER! Do not add your other text.)
9. **Submission Upload:** Use the dedicated upload link on MyTUTorD2L to upload your C code only. While multiple submissions are allowed, only the latest submission will be retained. If you upload the wrong file by mistake, simply re-upload the correct one, and the previous submission will be overwritten.
10. **Backup And Save:** Remember to save your work on the PC D:Drive and save regularly throughout the evaluation. In the event of PC malfunction or power failure, only 5 to 10 minutes (depending on the case) extra time will be allotted.

QUESTION:

Create a structured C program to develop a **Compound Measures - Pressure App**. The application must be able to calculate:

- **Unknown Force (F):**
Pressure (P) multiplied by Area (A)
- **Unknown Pressure (P):**
Force (F) divided by Area (A)
- **Unknown Area (A):**
Force (F) divided by Pressure (P)

1. Menu Driven Application

Within the main function, an **ELSE IF** selection structure must be used to determine user selections. The **ELSE IF** selection structure should be nested within (inside) a **DO WHILE** repetition structure, allowing the user to make multiple selections until they choose to exit. The **ELSE IF** selection structure must also determine invalid input and provide appropriate feedback.

Note: It is important to refer to the application print screens in this document as well as the given showcase .exe application to understand the functionality of the application and menu and to align your solution as closely as possible to it.

Important: The uploaded solution code may not contain a **SWITCH** selection structure. If a **SWITCH** is used, no marks will be awarded for the menu-driven functionality, even if the application works as it is supposed to.

2. Variables

Create the following variables at the beginning of the main function body and do not alter the name or type given:

- **char select** – used to capture and store the user selection.
- **float input1** and **float input2** – used to capture the user input for calculation.
- No other C-programming variables are needed or may be created in the application.

3. Heading

Display the following heading with your student number that will be displayed at the top of the application:

```
~~~~~  
Compound Measures - Pressure  
Std No: 123456789  
~~~~~
```

Figure 1. Heading Output

4. Menu Options

The below menu options must be displayed below the heading. The user must select an option by entering either 'a', 's', 'd', or 'f' characters.

```
a. Unknown Force (F)  
s. Unknown Pressure (P)  
d. Unknown Area (A)  
f. Exit  
Select:
```

Figure 2. Menu Option Output

5. Wait and Clear Screen

After the unknown parameter has been calculated and displayed, the application must wait for 3 seconds before clearing the screen and redisplaying the menu and options, only if the user did not select the exit option.

A **FOR** repetition structure must be used to repeat a 80-millisecond delay and display:

- **Sleep(80);** – create a program delay of 80 milliseconds.
- " ~" – display with each repetition.

To clear the screen, after the 3-second wait and display, the following statement must be used:

- **system("cls");** – clear the command prompt screen.

Note: To ensure that the **Sleep(80)** as well as the **system("cls")** statement functions as it should, the following libraries must be included below the standard input and output library:

- **#include <conio.h>**
- **#include <time.h>**

Adherence to the provided instructions, control structures, naming conventions, and structured programming principles is crucial for full marks, even if the program functions correctly. Proper indentation and comments are also essential for clarity and understanding. No global variables or self-written functions may be implemented in the final solution.

IMPLEMENT THE FOLLOWING FLOW FORMAT:

1. Libraries
2. Main function
 1. Variables
 2. Repetition Structure
 - i. Heading
 - ii. Menu Options
 - iii. Selection Structure
 - iv. Wait and Clear Screen

ANNEXURE A – EXAMPLE SCREEN OUTPUT

Menu Screen:

```
"C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. Memo\ES
Compound Measures - Pressure
Std No: 123456789

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:
```

Option 'a': Unknown Force

```
"C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. Memo\ES
Compound Measures - Pressure
Std No: 123456789

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:a

Input Pressure (P)):2.2
Input Area (A):4.16
Force = 2.20 * 4.16 = 9.15
```

Option 's': Unknown Pressure

```
"C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. Memo\ES
Compound Measures - Pressure
Std No: 123456789

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:s

Input Force (F):4.78
Input Area (A):1.84
Pressure = 4.78 / 1.84 = 2.60
```

Option 'd': Unknown Area

```
*C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. h
~~~~~
Compound Measures - Pressure
Std No: 123456789
~~~~~

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:d

Input Force (F):1.45
Input Pressure (P):2.14
Area = 1.45 / 2.14 = 0.68
~~~~~
```

Option 'f': Exit

```
*C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. h
~~~~~
Compound Measures - Pressure
Std No: 123456789
~~~~~

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:f

Bye Bye...
~~~~~
```

Invalid Input Selection

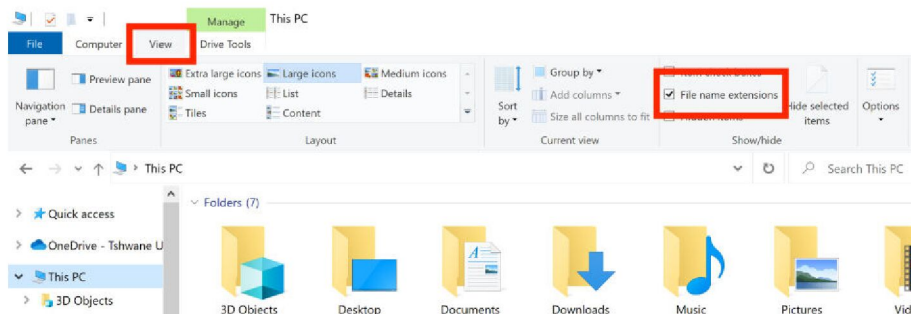
```
*C:\Users\EngelbrechtD\OneDrive - Tshwane University of Technology\TUT Academic\1. Teaching\1. ES216AB\2025\1. Evaluations\EV1\0. Memo\ES216AB_EV1B.e
~~~~~
Compound Measures - Pressure
Std No: 123456789
~~~~~

a. Unknown Force (F)
s. Unknown Pressure (P)
d. Unknown Area (A)
f. Exit
Select:v

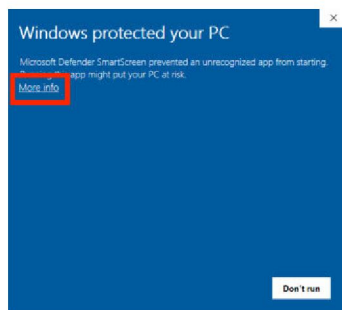
Invalid Selection...
~~~~~
```

ANNEXURE B – HOW TO RUN THE SHOWCASE

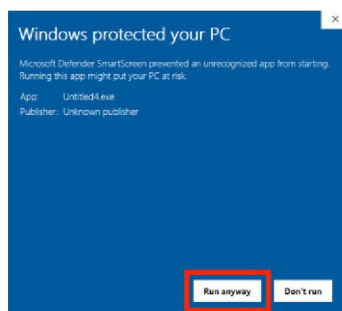
1. Enable “File name extensions” (see highlighted in red)



2. Change the file extension from “**Example.old**” to “**Example.exe**”
3. Run the “**Example.exe**” by double-clicking on the icon.
4. Windows may show the following. Click on “**More info**”



5. Click on “**Run anyway**”



ANNEXURE C – MARK ALLOCATION

Note: Score range is 0 - 4 which is: 0-none, 1-poor, 2-average, 3-good, 4-excellent

TEST RUBRIC	SCORE [0-4]	WEIGHT [%]
C CODE EVALUATION - Basic Logic		55
1. Overall Neatness & Readability		5
2. Menu Driven App: Do While & If Else		5
3. Heading: Display		5
4. User Selection: Display & Input		5
5. If Condition, Unknown 1, Input and Output		5
6. Else If Condition, Unknown 2, Input and Output		5
7. Else If Condition, Unknown 3, Input and Output		5
8. Else If Condition, App Exit		5
9. Else, Invalid Input		5
10.Wait & Clear, For Loop, Sleep, Cls		5
11. No Runtime or Compile Errors		5
TOTAL		55
STUDENT NUMBER		

Graduate Attribute	GA Number	GA Score [0-5]
Engineering Professionalism	GA10	
Application of scientific and engineering knowledge	GA2	
Engineering methods, skills, tools, including information technology	GA5	
Impact of Engineering Activity	GA7	

ANNEXURE D – INFORMATION SHEET

Libraries:	<stdio.h> , <stdlib.h> , <time.h> , <math.h>
Data types:	void, char, short, int, float, double
Data Type modifiers:	const, auto, static, unsigned, signed
Arithmetic operators:	* / % + -
Relational operators:	< <= > >= == !=
Assignment operator:	= += -= *= /= %= &= ^= = <<= >>=
Logic operators:	&& !
Bitwise logic operators:	& ^ ~ << >>
Pointer operators:	Dereference: * Address: &
Control Structures:	
IF Selection:	if (condition) { ... };
IF ELSE Selection:	if (condition) { ... } else { ... };
SWITCH Selection:	switch (control variable) { case 'value': ... ; break; default: ... ; break; }
FOR Loop:	for (initial value of control variable; loop condition; increment of control variable) { ... }
WHILE Loop:	while (condition) { ... };
DO WHILE loop:	do { ... } while (condition);
Functions:	return_data_type function_name (parameters) { ... };
Common Library Functions:	printf() , scanf() , rand() , srand() , time() , isalpha() , isdigit() , getchar() , getch() , strcpy()
Arrays:	One dimensional: data_type variable_name[size]; Two dimensional: data_type variable_name [x_size][y_size];

ANNEXURE E – ASCII TABLE

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL