

Laboratory Multimedia and Internet of Things Computer Engineering Department Institut Teknologi Sepuluh Nopember

Basic Programming Practicum

Introduction to CLanguage

2024

1 Goals

- Students can create projects within the IDE.
- Students can demonstrate their knowledge about program structure in the C language
- Students can demonstrate their knowledge about data types in the C language
- Students can demonstrate their knowledge of data types in the C language
- Students are able to use functions to read input from the keyboard
- Students are able to use functions to print text on the screen

2 Introduction to C Programming Language

The C language was developed by Dennis M. Ritchie and Brian W. Kernighan in the early 1970s. There are several standards for the C programming language. There are several guidelines for writing C programming language. Below are some of the standards:

- 1. Kernighan & Ritchie Definition(K&R)
- 2. ANSI-C (X-3.159 -1989-)
- 3. AT&T (for superset C, C++) definition, and
- 4. GNU Coding Standards

Implementation and uses of the C programming language

- 1. Creating operating systems and it's system programs
- 2. Programmin language that is "very close" to hardware (e.g., for device control).
- 3. Developing toolkits
- 4. Writing programs

3 IDE (Integrated Development Environment)

IDE stands for "Integrated Development Environment" in English. In Bahasa Indonesia, IDE can be translated as "Lingkungan Pengembangan Terintegrasi" or "Ruang Kerja Pengembangan Terpadu." IDE is a software designed to assist software developers in the process of development, coding, and testing computer applications.

Here are several list of C programming language IDE applications that can be used.

Code::Blocks

4 Creating new project in IDE Code::Blocks

4.1 Steps to create a new project

1. Go to File > New > Project

-	art here - Code:Blocks 20.03		Fortran wxSmith Tools Tools+ Plu	-	0	×
File	New	Ject Build Debug		nagina boxybobca securinga menp Shift-N Karaka ka Maraka ka Mar	c	
6	Open Open With Hex Editor Open default workspace	Ctrl+O	Class Project Build target			
	Recent projects Recent files	, ,	File Custom	Code::Blocks		
	Import project	>	From template Nassi Shneiderman diagram	The open source, cross-platform IDE		
8	Save file	Ctrl-S	Nassi Srineiderman diagram			
M	Save file as					
	Save project			Strand - State and State and State and State and State and State		
	Save project as		Rela	lease 20.03 rev 11983 (2020-03-12 18:24:30) gcc 8.1.0 Windows/unicode - 64 bit		
	Save project as template					
	Save workspace		_			
	Save workspace as			Create a new project Open an existing project V Tip of the Day		
-	Save everything	Ctrl-Shift-S				
0	Close file	Ctrl-W		Visit the Code::Blocks forums Report a bug or request a new feature		
	Close all files Close project	Ctrl-Shift-W				
	Close workspace			Recent projects		
	Print.	Ctrl-P		CUUserstieko myDocumentsiPraktikum01HaloDunia/Praktikum01HaloDunia.cbp		
-	Export			C/Usersleko my/Documentslys/xx.cbp		
	Properties			Recent files		
ത	Quit	Ctrl-Q		No recent files		
1		240-Q				
				@ 2004 - 2018, The Code::Blocks Team.		
	& others					×
Los	ruting: "C:\Program File	a\CodeBlocks/cb	cc X QBuild log X PBuild message: console runner.exe" "C:\Users\eko my" 0 minute(s), 3 second(s)}	ps x // Cpc/had/Wea++ x // Cpc/had/Wea+ masages X // Crope x @bobugger X // DayBlack x [Frotram info x & C gylocamenta/PraktikamDillaisDania/bin/Bebag/FraktikamDillaisDania.ese* (in Ci/Bersieko mylOcamenta/FraktikamDillaisDania/)	osed files	list > *
Crea	e a new project			default		

Figure 1

2. Click on Console Application

there X		Code::Blocks	
	New from temple	and the second s	×
		Category: <all categories=""></all>	Go
	Build targets Files Custom User templates	ARM AVR Arduino Code:El Console Project Project Project plugin applicat	Cancel
		Image: Second system Image: Se	
		Image: Second	
			sw as Large icons Útat
	TIP: Try right-clic	project project Project (share project project	
		- d type first on the left lic wizard from the main window (filter by categories if needed)	
		© 2004 - 2018, The <u>Code: Blocks</u> Team.	
5 others			

Figure 2

3. Choose C as the programming language

🔚 🎒 🐍 🦻 🖄 🐘 🏛 🔍 🔍 🖄 🕨 🗞 🐼 🖬 Debug	- 등 ▶ 년 만 일 값 다 년 표 표 표 2 []	
ere X		
	States wanted a state of the st	
	Code::Blocks	
	and a start and a start	
	Console application >	C C C C C C C C C C C C C C C C C C C
	Console Please select the language you want to use.	
	Please make a selection	
	C++	
	Re	24
		the Day
	1	<u>e</u>
	A	cbp
	4	
	< Back Next > Cancel	
	© 2004 - 2018, The Code::Blocks Team.	
Thers		
ode::Blocks × 🔍 Search results × 📝 Cocc × 🔅 Build 🖡	g 🗴 🕈 Build messages 🗶 📝 CppCheck/Vera++ 🗶 📝 CppCheck/Vera++ messages 🗴 📝	Cscope X ODebugger X DoxyBlocks X F Fortran info X & Closed file:
ing: "C:\Program Files\CodeBlocks/cb console runner. s terminated with status -1073741510 (0 minute(s), 3	xe" *C:\Users\eko my\Documents\Praktikum0HaloDunia\bin\Debug\Praktikum0HaloDuni	la.exe" (in C:\Users\eko my\Documents\Praktikum0lHaloDunia\.)
		default
re		

Figure 3

4. Insert your project name

- /***<	⇔⇒∠ಾ⊛.∗ ⊧ ⊏≌⊿		C 📃 🗸 🔍	<u> </u>		
		application	. · · ·	x set		
	4			cbp		
		< Back	Next > Car	cel		
		© 2004 - 2018, The Code::Bl	cos ream.			
nters Code:Blocks X	unner.exe" "C:\Users\eko my\Docume	✓ CppCheck/Vera++ ※ ✓ CppChec ents\Praktikum01HaloDunia\bin\De	k/Vera++ messages 3 oug\Praktikum01Bal	Cscope × Obebugger × Dunia.exe* (in C:\Users\eko my	DoxyBlocks X F Fortran info X Clos Documents\Praktikum01BaloDunia\.)	ed fib

Figure 4

5. Choose the compiler (gcc), select a directory to save your project, and click save

		› Plugins Doxy®locks Settings Help ■ ▶ 極像電話時間間 ■ ■ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	
\$/**< 0 ? < 0 0 > D			
here X			
		Code::Blocks	
		Console application X	
		Console You want enabled in your project. Compilar:	
		Rs GNU GCC Compiler 2	
		Create "Debug" configuration: Debug	
		"Debug" options	
		Output dir.: bin\Debug\ E	
		Objects output dir: obj\Debug\	
		Create "Release" configuration: Release	
		"Release" options cbp	
	•	Output dir: bin\Release\	
		Objects output dir.: obj/Release\	
		< Back Finish Cancel	
		© 2004 - 2018, The Code::Blocks Team.	
others			
Code:Blocks × Search results × Cccc	× \$Build log × PBuild me sole_runner.exe* "Cr\Usera\e nimute(s), 3 second(s)}	ssages X 🖉 CopcHeck/Vera++ X 🖉 CopcHeck/Vera++ messages X 🖉 Cocope X 🖉 Debugger X 🖉 DoxyBicks X 🖡 Fortma info X 🀁 Closed Ko my/bocumente/Prattikm0)Halobuniebun/Debug/Prattikm0/Halobunia.exe* (In Ci/Derra/eko my/Documente/Prattikm0)Halobunia.)	l files

Figure 5

6. Write code as in Figure 6 in Code::Blocks

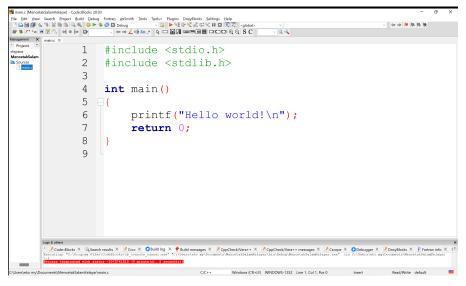


Figure 6

7. Click Build- >Build and Run or press F9 on your keyboard

ragement X main.c X	Compile current file Ctrl-Shift-F9 Run Ctrl-F10			
Projects pace	Build and run F9 Rebuild Ctrl-F11			
Sources	Clean	e <stdlib.h></stdlib.h>		
main.c	Build workspace Rebuild workspace Clean workspace			
	Abort	n ()		
	Errors	>		
	Select target	· · · · · · · · · · · · · · · · · · ·		
	Export Makefile	<pre>ntf("Hello world!\n");</pre>		
	8 } 9 -			

Figure 7

8. The program output can be seen on the console tab

<pre>####################################</pre>	🏮 /** *< 💿 ? 🔇					★ # # # ← ← ↓
<pre>* 1 #include <stdio.h> active ac</stdio.h></pre>						
verse maine 2 #include <stdlib.h> ■"CVDerrike myDoursetti MenotalisatemKalgarbenDebug/MenotalisatemKalgarase" - □ X Milo world! Morcess returned 0 (0x0) execution time : 0.023 s</stdlib.h>	sce	1 #include <s< th=""><th>stdio.h></th><th></th><th></th><th></th></s<>	stdio.h>			
Hello world! Process returned 0 (0x0) execution time : 0.023 s	cetakSalam ources main.c	2 #include <s< td=""><td>stdlib.h></td><td></td><td></td><td></td></s<>	stdlib.h>			
Process returned 0 (0x0) execution time : 0.023 s			\Debug\MencetakSalamKelayar.exe*	-	• ×	
			0.423 s		-	
	Loss & other		···· · · · · · · · · · · · · · · · · ·	Conclusion X 2 c		
* 2 CodeBlock X QuSench results X 2 Cocc X @Build tog X * Build messages X 2 CpcDect/Vers++ X 2 CpcDect/Vers++ messages X 2 Cocco X @Debugger X 2 DonyBlocks X * Fortrain in Uncreased; UnablockedPace2010;ersilves mytogenetations under the Unit of Unit	\Microso Executin	de:Blocks X Search results X / Cccc X PBuild log X / ft/WindowaApps:Cl/Usersleko my\AppBats\Local\Programs\MiRt g: "Cl\Program Files\CodeBlocks/ob_console_rummer.exe" "Cl	Bund messages A / CppCheck/Vera++ A TeXinitex/binix64 :\Users\eko my\Documents\MencetakBalamKelay	<pre>cppunecs/vera++ messages ~ 2 cl ar\bin\Debug\MencetakSalanKelayar.c</pre>	exe" (in Ci\Users\eko m	/\Documents\MencetakSalamKela

Figure 8

4.2 Pre-lab Assignment

- 1. Try looking for another IDE besides codeblocks, and explain the advantages and disadvantages of each IDE!
- 2. What makes C different from other programming languages? explain the advantages and disadvantages!
- 3. Create a project with name HaloDunia and write the program like can be seen on figure 6 but change Hello World! with other words!

5 Structure of C Programming Language

```
1 #include <stdio.h>
2
3 int main()
4 {
5 //printing to screen
6 printf("Halo World");
7 return 0;
8 }
```

Code on Listing 1 is a simple program to print "Halo Dunia" to screen. The following is the explanation what each line of code do in the program.

Row 1 : #include <stdio.h>

header file library for input and output functions like printf() (the one used on line 6)

- Row 2 : Empty line.
- Row 3: int main()

The main function. The main function is the first function to be ran when the program starts.

```
Row 4 : {
```

Beginning of the main() function code block.

Row 5: //printing to screen

Comments. Comments are used to explain what the program is doing. Comments are ignored by the program, but helps the reader.

- Row 6: printf("Halo Dunia"); Printing "Halo Dunia" to the screen.
- Row 7: return 0;

Returning the main() function (A function ends when it returns)

Row 8 : }

Closing the main() function code block.

5.1 Pre-lab Assignment

- 1. Try to swap line 6 and line 7 in Listing 1. Explain what happened?
- 2. What if return 0; replaced with return 1;?
- 3. What happens if return 0; is deleted?

6 Data Types and Variable

6.1 Data Types

In C programming language, there are several data types to represent integer, real number, characters, string, and etc.

Table 1: Some data types in C programming language

Data Types	Size	Range Value
Int (or signed int)	2 bytes	-32,768 to 32,767
unsigned int	2 bytes	0 to 65,535
Short int(or signed short int)	2 bytes	-32,768 to 32,767
Long(or singed short int)	4 bytes	-2,147,483,648 to 2,147,483,647
unsigned long	4 bytes	0 to 4,294,967,295
float	4 bytes	1.2E-38 to 3.4E+38
double	8 bytes	2.3E-308 to 1.7E+308
Long double	10 bytes	3.4E-4932 to 1.1E+4932
char(or signed char)	1 byte	-128 to 127
unsigned char	1 byte	0 to 255

To show the data on screen, every data type has a format specifier that can be used on formatted string. The following is the format specifier for several data types.

Format Specifier	Data Types			
%d or %i	int			
%f	float			
%lf	double			
%с	char			
%s	string			

Table 2:	Format	Specifier
----------	--------	-----------

There are still more data types that what was written on Table 1. These data types and its specification can be found easily on the internet.

6.1.1 Modifier

In general, a modifier is a word, phrase, or clause that modifies or describes a noun or verb in a sentence. Meanwhile, in programming languages, a modifier is a keyword used to alter the behavior or characteristics of an element in a program, such as variables, functions, or classes. We use modifiers to change the range of basic data types to fit programming needs. There are four modifiers, namely:

1. signed

int value = -10; (Can store integer type variables for negative and positive values)

2. unsigned

unsigned int count = 100; (Can stroe integer type variables for positive value only)

3. long

long population = 7500000000; (Using the long data type to store values larger than the int data type.)

4. short

short temperature = 20; (Using the short data type to save memory when we know that the values to be stored will be relatively small.)

6.2 Variable

1

Variables are places to store data. Declaring a variable can be done in the following ways

Listing 2: C variable declaration

```
DataType VariableName;
```

6.2.1 Aithmetic and Assignment Operator

Operators are symbols that represent operations to be performed on one or more operands. Arithmetic Operators can be used to perform arithmetic operations such as addition, subtraction, etc. The Assignment Operator can be used to perform an operation on a variable and change the value of the variable according to the results of the operation.

Operator	Name	Example
+	Addition	x + y
-	Subtraction	x = y
*	Multiplication	x * y
/	Distribution	x/y
%	Modulo	х % у

Table 3: Arithmetic operator in C programming language

Table 4: Assignment operator

Operator	Example	Similiar meaning
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
&=	x &= 3	x = x & 3
=	x = 3	x = x 3
^=	x ^= 3	x = x ^3
>>=	x >>= 3	x = x >>3
<<=	x <<= 3	x = x <<3

There are also 'abbreviations' for some assignment operators such as x+=1 and x-=1, which ++ and --.

x++;

x--;

++x;

--x;

6.2.2 Operator Bitwise

Bitwise operators are special operators used to handle logical operations on binary numbers in bit form. Binary numbers themselves are a type of number consisting of only two digits, which are 0 and 1. If the original value used is not in binary, it will be automatically converted by the C compiler into a binary number. For example, 7 in decimal equals 0111 in binary

Operator	Name	Example	Binary	Result (binary)	Result (decimal)		
&	AND	10 & 12	1010 & 1100	1000	8		
	OR	10 12	1010 1100	1110	14		
٨	XOR	10 ^12	1010 ^1100	0110	6		
~	NOT	~10	~1010	0101	-11 (two complement)		
<<	Left shift	10 <<1	1010 <<1	10100	20		
>>	Right shift	10 >>1	1010 >>1	101	5		

Table 5: Bitwise operator

Notes: There are several operators in the C language. Please study them by seeking references independently

6.3 Pre-lab Assignment

Listing 3:	Using ass	ignment op	perator in a	const variable
------------	-----------	------------	--------------	----------------

```
1
  #include <stdio.h>
2
  int main()
3
  {
4
5
     //variable declaration
6
      const int x=0;
7
       x=1;
8
       printf(x);
9
       return 0;
10
  }
```

- 1. Try to compile the program in Listing 3, what happened?
- 2. What must be done so that the output of Listing 3 is 1?

Listing 4: Using format specifier

```
1 #include <stdio.h>
2 int main(){
3    float a = 3.14;
4    printf("%d", a);
5 }
```

- 3. Try to compile the program in Listing 4, what happened?
- 4. What must be done so that the program in Listing 4 has a score of 3.14?

7 Input and Output

7.1 printf()

printf is a function in C that is used to print formatted string. You can use format specifier % within the formatted string to outputs your variables.

printf(const char *format,v1,v2,..,vn)

The format specifier for each data types can be seen on Table 2

Contoh 7.1.1 Printing text to the screen.

Listing 5: Print text "C Programming" Ke layar

```
1 #include <stdio.h>
2 int main()
3 {
4 // Printing text inside the " symbol
5 printf("C Programming");
6 return 0;
7 }
8
```

- All C program must have main() function where the program needs to run the code.
- printf() function is a function from stdio.h library. This function outputs the string inside the symbol "" to the screen.
- return_0; statement in the main() function tells the program to exit.

Contoh 7.1.2 Printing integer.

```
1 #include <stdio.h>
2 int main()
3 {
4 int testInteger = 5;
5 printf("Number = %d", testInteger); // <- %d format string
6 return 0;
7 }
8
```

The code above uses the format specifier %d to prints int data type. The %d part of the string will be replaced with the value of testInteger.

Contoh 7.1.3 Real number output (float atau double)

- Base : using float data type.
- Height: using float data type.
- Area: using float data type.

$$Area = \frac{1}{2} \times Base \times Height \tag{1}$$

```
1
       #include <stdio.h>
 2
3
       int main()
4
       ſ
5
         // variable declaration
6
         float Base;
 7
         float Height;
8
         float Area;
9
         // value initialization
10
         Base = 10;
11
         Height = 5;
12
         // calculating area
13
         Area = 0.5*Base*Height;
14
         // printing the text to screen
15
         printf("Area = %f",Area);
16
         return 0;
17
       }
18
19
```

explanation

Row 6-8 Base, Height and Area are floatdata type which use to store triangle area data.

Row 10 dan 11 Value initialization to Base=10 and Height=5

Row 13 Calculating triangle area according to the equation 7.1

Row 15 Printing Area to the screen using printf command.

7.2 scanf

Function $scanf(const_{\Box}char_{\Box}*format,_{\Box}...)$ reads input according to the format string.

1. Syntax

scanf(const char *format, ...)

2. Parameter

Format string in C consist of one or more whitespace, non-whitespace, and format specifiers.

3. Return Value

The function will return the value of arguments it has sucessfully read.

Example 7.2.4 Calculating triange Base dan tinggi Height yang diinputkan dari keyboard.

```
1 #include <stdio.h>
2
3 int main()
4 {
5 float Base, Height, Area;
```

```
6
7
     printf("Calculate triangle area\n");
8
     printf("\Insert Base= ");
9
     scanf("%f",&Base);
10
     printf("\nMasukkan Height=");
     scanf("%f",&Height);
11
12
     Area = 0.5*Base *Height;
13
     printf("Triangle Area = %.2f", Area);
14
     return 0;
15 }
16
```



Figure 9

Row 9 scanf("%f",&Area); requesting input for triangle base

Row 11 scanf("%f",&Height); requesting input for triangle height

Row 13 printf("Triangle Area = %.2f", Area);, .2 in %.2f indicating that only 2 digits after the decimal point need to be printed

Example 7.2.5 Program to input name and email.

This example shows how to input string or text from keyboard and outputs it on the screen. Input from this program consist of sName and sEmail. Because the text contains many characters, each variable is declared as an array of characters with the number of characters for sName=20 and sEmailAddress=30.

```
1
       #include <stdio.h>
 2
3
       int main ()
 4
       ſ
 5
         char sName[20], sEmail[30];
 6
 7
         printf("Enter Name: ");
8
         scanf("%19s", sName);
9
10
         printf("Enter Email: ");
11
         scanf("%29s", sEmail);
12
13
         printf("Name : %s\n", sName);
14
         printf("Email:%s", sEmail);
         return(0);
15
       }
16
17
```

7.3 Escape Sequence

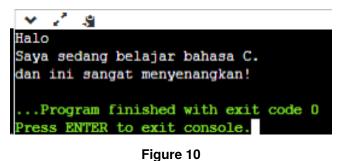
Some characters can't be written on the format string because they are used to format the outputs. So, to outputs those special characters we use escape sequences.

Escape sequence	Output
\a	Bell, alarm
\b	Backspace
\f	Change Page
∖n	Change Row
\r	Carriage return
\t	Tab Horizontal
\v	Tab Vertikal
\'	Single Quotes
\"	Double Quotes
\?	Question Mark
//	Backslash

Table 6	6: Escape	Sequence
---------	-----------	----------

Contoh 7.3.6 Change row with escape sequence \n.

```
1 #include <stdio.h>
2
3 int main()
4 {
5 printf("Halo \nI'm learning C programming language.\nand it's so fun!");
6 return 0;
7 }
8
```



i iguio i

Contoh 7.3.7 Using escape sequence \t to change tab.

```
#include <stdio.h>
 1
2 int main(void)
3 {
4
   printf("Name \t\t: Rahmad Rahardi\n");
5
    printf("Address \t\t: Bendungan Hilir Jakarta\n");
6
    printf("Place of Birth \t: Jakarta\n");
7
    printf("Date of Birth \t: 30 February 2000\n");
8
9
    return (0);
10 }
```

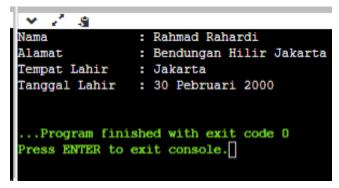


Figure 11

7.4 Pre-lab Assignment

- 1. scanf("%d", &height), the code will ask for input to be entered in the variable height. Explain why there is "&" on &height! what happens if it is simply written as height whitout &!
- 2. Try to build a program to read input for the name and NRP and display it to the screen!
- 3. Try to build a program that asks user to enter a number, then displays the squared dan cubed value of it!

Optional: Learn Git and Github. You can start learning from the following resources: GitHub - https://github.com Git - https://git-scm.com/doc