

LAPORAN PRAKTIKUM  
PEMROGRAMAN TERSTRUKTUR  
PRAKTIKUM I – PENDAHULUAN  
KELAS B



Disusun Oleh:

Nama: Andro Syahreza

NIM: 175090807111007

Hari/Tgl. Praktikum: Selasa, 9 April 2019

LABORATORIUM KOMPUTASI  
JURUSAN FISIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS BRAWIJAYA  
2019

## A. Percobaan 1:

<p><b>Task</b></p> <p>This challenge requires you to print <i>Hello, World!</i> on a single line, and then print the already provided input string to <code>stdout</code>.</p> <p>Note: You do not need to read any input in this challenge.</p>
<p><b>Input Format</b></p> <p>You do not need to read any input in this challenge.</p> <p><b>Sample Input 0</b></p> <pre>Welcome to C programming.</pre>
<p><b>Output Format</b></p> <p>Print <i>Hello, World!</i> on the first line, and the string from the given input on the second line.</p> <p><b>Sample Output 0</b></p> <pre>Hello, World! Welcome to C programming.</pre>

## Solusi :

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <math.h>
4  #include <stdlib.h>
5
6  int main()
7  {
8
9      char s[100];
10     scanf("%[^\\n]*c", &s);
11
12     printf("\\nHello World!\\n");
13     printf("%s", s);
14
15     return 0;
16 }
```

Gambar 1.1 (Kode Program Percobaan 1)

```
Welcome to C programming.

Hello World!
Welcome to C Programming.
```

Gambar 1.2 (Keluaran dari Program)

## Penjelasan :

Pada program 1 (Gambar 1.1) masukan user menggunakan perintah `scanf` yang akan disimpan pada variable 's'. Variable 's' beripe string dengan panjang string sebanyak 100 character. Percobaan tersebut menginginkan keluaran cetakan menjadi 2 line, line pertama adalah "Hello World!" dan line kedua adalah "Welcome to C Programming.". `[\n]` pada perintah `scanf` berarti character akan diperiksa sampai user mengetik tombol enter, hal ini memungkinkan user untuk dapat mengetik kalimat termasuk dengan spasi dalam masukan. Untuk mencetak "Hello World!" digunakan perintah `printf` dengan sequence '\n' untuk memberikan line baru. Masukan user akan dicetak dengan perintah `printf`, format string '%s' akan digantikan dengan variable 's'. Gambar 1.2 menunjukkan keluaran yang dihasilkan sesuai dengan kode program yang diinginkan pada percobaan 1.

## B. Percobaan 2:

<b>Task</b> You have to print the character, <i>ch</i> , in the first line. Then print <i>s</i> in next line. In the last line print the sentence, <i>sen</i> .	
<b>Input Format</b>  First, take a character, <i>ch</i> as input. Then take the string, <i>s</i> as input. Lastly, take the sentence <i>sen</i> as input.	<b>Sample Input 0</b>  <pre>C Language Welcome To C!!</pre>
<b>Output Format</b>  Print three lines of output. The first line prints the character, <i>ch</i> . The second line prints the string, <i>s</i> . The third line prints the sentence, <i>sen</i> .	<b>Sample Output 0</b>  <pre>C Language Welcome To C!!</pre>

**Solusi :**

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <math.h>
4  #include <stdlib.h>
5
6  int main() {
7      char ch;
8      char s[10];
9      char sen[100];
10
11     scanf("%c\n", &ch);
12     scanf("%s\n", s);
13     scanf("%99[^\n]%*c", sen);
14
15     printf("\n%c\n", ch);
16     printf("%s\n", s);
17     printf("%s\n", sen);
18
19     return 0;
20 }
```

*Gambar 2.1 (Kode Program Percobaan 2)*

```
C
Language
Welcome To C!!

C
Language
Welcome To C!!
```

*Gambar 2.2 (Keluaran dari Program)*

**Penjelasan :**

Pada program 2 (Gambar 2.1) inisiasi 'char ch;' adalah untuk tipe variabel character, lalu pada line 8 's[10]' pada char menandakan inisiasi string dengan panjang character sebanyak 10 buah. Hal yang sama dilakukan untuk string yang lebih panjang yaitu pada line 9 'sen[100]'. Masukan user menggunakan perintah scanf yang akan disimpan pada variabel ch, s, dan sen. [^\n] pada perintah scanf berarti character akan diperiksa sampai user mengetik tombol enter, hal ini memungkinkan user untuk dapat mengetik kalimat termasuk dengan spasi dalam masukannya. Percobaan 2 tersebut menginginkan keluaran berupa character, string, dan kalimat. Gambar 2.2 menunjukkan hasil keluaran sama persis dengan masukan yang diberikan.


### C. Percobaan 3:

<b>Task</b> <p>Your task is to take two numbers of int data type, two numbers of float data type as input and output their sum:</p> <ol style="list-style-type: none"><li>1. Declare <b>4</b> variables: two of type int and two of type float.</li><li>2. Read <b>2</b> lines of input from stdin (according to the sequence given in the 'Input Format' section below) and initialize your <b>4</b> variables.</li><li>3. Use the + and - operator to perform the following operations:<ul style="list-style-type: none"><li>◦ Print the sum and difference of two int variable on a new line.</li><li>◦ Print the sum and difference of two float variable rounded to one decimal place on a new line.</li></ul></li></ol>	
<b>Input Format</b> <p>The first line contains two integers.</p> <p>The second line contains two floating point numbers.</p>	<b>Sample Input</b> <pre>10 4 4.0 2.0</pre>
<b>Output Format</b> <p>Print the sum and difference of both integers separated by a space on the first line, and the sum and difference of both float (scaled to 1 decimal place) separated by a space on the second line.</p> <p><b>Sample Output</b></p> <pre>14 6 6.0 2.0</pre>	

### Solusi :

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <math.h>
4  #include <stdlib.h>
5
6  int main()
7  {
8      int a,b;
9      float c,d;
10
11     scanf("%d %d", &a, &b);
12     scanf("%f %f", &c, &d);
13
14     printf("%d %d\n", a+b, a-b);
15     printf("%.1f %.1f", c+d, c-d);
16
17     return 0;
18 }
```

Gambar 3.1 (Kode Program Percobaan 3)



```
10 4
4.0 2.0
14 6
6.0 2.0
```

*Gambar 3.2 (Keluaran dari Program)\*

**Penjelasan :**

Pada program 3 (Gambar 3.1) terdapat 4 variabel, 2 bertipe integer dan 2 bertipe float. Masukan user menggunakan perintah scanf yang akan disimpan pada variabel a dan b untuk tipe integer. Juga perintah scanf akan disimpan pada variabel c dan d untuk tipe float. Keluaran yang diinginkan adalah hasil pengurangan dan penjumlahan dari masing-masing variabel yang bertipe sama. Untuk keluaran line pertama format ‘%d’ atau integer akan digantikan oleh variabel a + b dan a-b. Untuk keluaran line kedua format ‘%f’ atau float akan digantikan oleh variabel c + d dan c-d. Gambar 3.2 menunjukan keluaran yang dihasilkan sesuai dengan kode program yang diinginkan pada percobaan 3.