Nama : Muhammad Rafi Cahya Ramadhana

NIM : 22/492162/PA/21075

# WEEK 2 ASSIGNMENT

## Problem 1

Create a program that computes the elapsed time based on the velocity and the distance from the user's input!

For example, if the program takes the input the velocity of 40, dan the distance of 100, then output the value of 2.5.

## Source code

```
#include<iostream>
using namespace std;

int main() {
    //Declare the variables
    double v, d, t;

    //Input values to variables
    cout << "Input the velocity (in km/h) \n";
    cin >> v;
    cout << "Input the distance (in km) \n";
    cin >> d;

    //Calculate the elapsed time
    t = d/v;

    //Print the result
    cout << "Elapsed time is  " << t << " hour";


    return 0;
}
```

## Screenshot

```
Input the velocity (in km/h)
40
Input the distance (in km)
100
Elapsed time is  2.5 hour
Process finished with exit code 0
```

```
Input the velocity (in km/h)
60
Input the distance (in km)
300
Elapsed time is  5 hour
Process finished with exit code 0
```

## Penjelasan

We can easily divide distance by velocity to get the value of time.

## Problem 2

Create a program that reads the two data in form of time, then computes the difference between the data! The dialogue is as the following:

- Input the 1st hour data: …
- Input the 1st minute data: …
- Input the 2nd hour data: …
- Input the 2nd minute data: …
- Output: The difference between the two times is … minutes

Assume that the first time is always earlier than the second time.

## Source code

```
#include<iostream>
using namespace std;

int main() {
    //Declare the variables
    int a,b,c,d;
    int first,second,diff;

    //Input values to variables
    cout << "Input the first hour data \n";
    cin >> a;
    cout << "Input the first minute data \n";
    cin >> b;
    cout << "Input the second hour data \n";
    cin >> c;
    cout << "Input the second minute data \n";
    cin >> d;

    //Calculate the time difference
    first = a*60 + b;
    second = c*60 + d;

    diff = abs(first-second);

    //Print the result
    cout << "The difference between the two times is " << diff << " minutes";


    return 0;
}
```

## Screenshot

```
Input the first hour data
11
Input the first minute data
30
Input the second hour data
13
Input the second minute data
10
The difference between the two times is 100 minutes
Process finished with exit code 0
```

```
Input the first hour data
7
Input the first minute data
30
Input the second hour data
10
Input the second minute data
00
The difference between the two times is 150 minutes
Process finished with exit code 0
```

Penjelasan

First, we need to convert the hour value to minute and then subtract the second time by the first time, and we should get the time difference in minute.

## Problem 3

> Create a program that read two numbers, then prints those two numbers in reverse!
>
> For example, if the input is "15 4" (without quotation mark), then the output is "4 15" (without quotation mark).

Source code

```cpp
#include<iostream>
using namespace std;

int main() {
    //Declare the variables
    double a,b;

    //Input values to variables
    cout << "Input the numbers \n";
    cin >> a >> b;

    //Show the result
    cout << b << " " << a;

    return 0;
}
```

Screenshot

```
Input the numbers
15 4
4 15
Process finished with exit code 0
```

```
Input the numbers
26 12
12 26
Process finished with exit code 0
```

Penjelasan

We can easily swap the position of variables a and b in the `cout` syntax to get the output value swapped.

## Problem 4

Create a program that reads a positive number, then prints the largest integer less than or equal to the input!

For example, if the input is "9.4562" (without quotation mark), then the output is "9" (without quotation mark). If the input is "10" (without quotation mark), then the output is "10" (without quotation mark).

## Source code

```cpp
#include<iostream>
using namespace std;

int main(){
    //Declare the variable
    int x;

    //Input value to variable
    cout << "Input number \n";
    cin >> x;

    //Show the result
    cout << x;

    return 0;
}
```

## Screenshot

```
Input number
9.4562
9
Process finished with exit code 0
```

```
Input number
10
10
Process finished with exit code 0
```

```
Input number
22.342156247313216781 32
22
Process finished with exit code 0
```

## Penjelasan

We can simply use the data type `int` if we want to use bilangan bulat. The computer will ignore all the decimals because the data type `int` can only contain integer (bilangan bulat) value.

## Problem 5

Given a sequence: 1, 3, 6, 10, 15, 21, 28, 36, ...

- Determine the formula of given sequence!
- Create a program that reads a number n, and prints the n-th term of given sequence! For example, if the input is 3, then the input is 6.
- Try various inputs: 10, 100, 1000, 10000, 100000, 1000000. If the input is 100000 or 1000000, does the output match your expectation? If no, how does that happen? How to solve it so that the output match your expectation?

## Source code

```cpp
#include<iostream>
using namespace std;

int main(){
    //Declare the variables
    int n,x;

    //Input the number n
    cin >> n;

    //Calculate the nth number of the sequence using for loop
    for (int i; i<(n+1); i++) {
        x += i;

    }

    //Show the result
    cout << x;

    return 0;
}
```

## Screenshot

```
3
6
Process finished with exit code 0
```

```
10
55
Process finished with exit code 0
```

```
100
5050
Process finished with exit code 0
```

```
1000
500500
Process finished with exit code 0
```

```
10000
50005000
Process finished with exit code 0
```

Penjelasan

- Determine the formula of given sequence!

The formula to find $U_n$ is $U_{(n-1)} + n$

Or we can also say the previous number added by n

This is because the pattern of the given sequence is A (the origin) = $U_1$, $U_2$ ( $U_1 + 2$),

$U_3$ ( $U_2 + 3$), $U_4$ ( $U_3 + 4$), and so on.

- Create a program that reads a number n, and prints the n-th term of given sequence! For example, if the input is 3, then the input is 6.

To create a program that can return the $n^{th}$ number of the given sequence, we can use for loop to continuosly add the previous numbers and n until we get the desired $U_n$.

- Try various inputs: 10, 100, 1000, 10000, 100000, 1000000. If the input is 100000 or 1000000, does the output match your expectation? If no, how does that happen? How to solve it so that the output match your expectation?

All inputs satisfy the expected outputs except `100000` and `1000000`,

```
100000
705082704
Process finished with exit code 0
```

```
1000000
1784293664
Process finished with exit code 0
```

this is because the data type used (`int`) cannot contain such large number, it causes the number to loop back. In order to solve the issue, we can try change the data type to something that could contain more, namely `double`.

```
//Declare the variables
double n,x;
```

The new outputs for `100000` and `1000000`

```
100000
5.00005e+09
Process finished with exit code 0
```

```
1000000
5e+11
Process finished with exit code 0
```

Although it doesn't directly show the expected value, we can still derive the expected value from it