

Nama : Ifham Syafwan Fikri
Github : <https://github.com/FikriSyafwan/Penugasan-Praktikum-Pemrograman>
Nomor Mahasiswa : 24/545184/PA/23161
Kelas : KOM B
Dosen Pengampu : Muhammad Husni Santriaji

LAPORAN PRAKTIKUM PEMROGRAMAN

PERTEMUAN 2

Number 1: Creating payslip for an employee

Pada pertemuan kedua, salah satu masalah yang diangkat adalah membuat slip gaji karyawan dari suatu perusahaan dengan beberapa variable, yaitu nama, gaji kotor, pajak, angsuran atau cicilan, asuransi, dan terakhir gaji bersih. Untuk membuat program yang menghasilkan slip gaji yang diinginkan, pertama diperlukan deklarasi fungsi sebagai berikut:

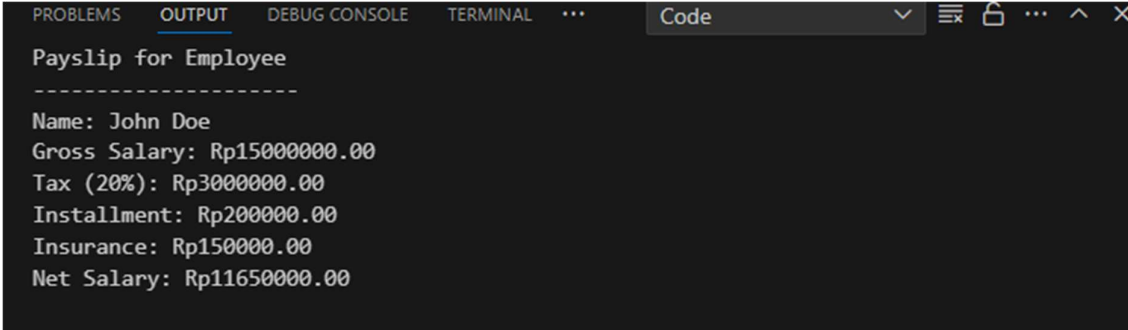
```
6 void generatePayslip(string name, double baseSalary, double percentage,  
7 double installment, double insurance) {  
8     // Calculate Gross Salary  
9     double grossSalary = baseSalary * (percentage / 100.0);  
10  
11     // Calculate Tax  
12     double tax = grossSalary * 0.20;  
13  
14     // Calculate Net Salary  
15     double netSalary = grossSalary - tax - installment - insurance;  
16  
17     // Output the payslip  
18     cout << fixed << setprecision(2);  
19     cout << "Payslip for Employee" << endl;  
20     cout << "-----" << endl;  
21     cout << "Name: " << name << endl;  
22     cout << "Gross Salary: Rp" << grossSalary << endl;  
23     cout << "Tax (20%): Rp" << tax << endl;  
24     cout << "Installment: Rp" << installment << endl;  
25     cout << "Insurance: Rp" << insurance << endl;  
26     cout << "Net Salary: Rp" << netSalary << endl;  
27 }  
28
```

Fungsi tanpa nilai kembalian, dengan nama “generatePayslip”, akan menghitung *variable* nilai dari gaji karyawan yang memiliki parameter berupa

1. String name
2. Double baseSalary
3. Double percentage
4. Double installment
5. Double insurance

Karena fungsi merupakan fungsi tanpa nilai kembalian, *output* akan dituliskan dalam fungsi, dengan sintaks “fixed << setprecision(2)” memberikan dua angka di belakang koma pada nilai *variable*. Setelah itu, program akan mengeluarkan *variable* yang telah diberi nilai dan mengeluarkan fungsi dengan parameternya telah diberi nilai sebagai berikut:

```
29 int main() {
30     string name = "John Doe";
31     double baseSalary = 15000000.00; // Base salary
32     double percentage = 100; // 100% of the base salary
33     double installment = 200000.00;
34     double insurance = 150000.00;
35
36     generatePayslip(name, baseSalary, percentage, installment, insurance);
37
38     return 0;
39 }
```



The screenshot shows the 'OUTPUT' tab of a code editor. The output text is as follows:

```
Payslip for Employee
-----
Name: John Doe
Gross Salary: Rp15000000.00
Tax (20%): Rp3000000.00
Installment: Rp200000.00
Insurance: Rp150000.00
Net Salary: Rp11650000.00
```

Number 2: Solving a quadratic equation

Program ini menghitung akar-akar dari suatu persamaan kuadrat yang diseleksi menjadi dua akar riil (memotong sumbu-x), satu akar riil (menyinggung sumbu-x), dan tidak memiliki akar riil sama sekali (tidak memotong dan menyinggung). Nilai-nilai dari koefisien persamaan kuadrat akan dituliskan sebagai berikut:

```

6  int main() {
7      double a, b, c;
8      double discriminant, root1, root2, realPart, imaginaryPart;
9
10     // Input coefficients a, b, c
11     a = 1;
12     b = 2;
13     c = 3;
14
15     // Calculate the discriminant
16     discriminant = b * b - 4 * a * c;

```

Deklarasi berupa nilai koefisien a, b, dan c yang dilanjutkan dengan deklarasi

1. double discriminant
2. double root1
3. double root2
4. double imaginaryPart

diperlukan untuk menghasilkan nilai akar-akar dari persamaan kuadrat. Sebagai contoh nilai koefisien a diberi nilai 1, b nilai 2, dan c nilai 3.

```

15     // Calculate the discriminant
16     discriminant = b * b - 4 * a * c;
17
18     // Determine the nature of the roots based on the discriminant
19     if (discriminant > 0) {
20         // Two distinct real roots
21         root1 = (-b + sqrt(discriminant)) / (2 * a);
22         root2 = (-b - sqrt(discriminant)) / (2 * a);
23         cout << "Roots are real and different." << endl;
24         cout << "Root 1 = " << root1 << endl;
25         cout << "Root 2 = " << root2 << endl;
26     }
27     else if (discriminant == 0) {
28         // One real root
29         root1 = -b / (2 * a);
30         cout << "Roots are real and the same." << endl;
31         cout << "Root = " << root1 << endl;
32     }
33     else {
34         // Complex roots
35         realPart = -b / (2 * a);
36         imaginaryPart = sqrt(-discriminant) / (2 * a);
37         cout << "Roots are complex and different." << endl;
38         cout << "Root 1 = " << realPart << " + " << imaginaryPart << "i" << endl;
39         cout << "Root 2 = " << realPart << " - " << imaginaryPart << "i" << endl;
40     }
41
42     return 0;
43 }

```

Setelah itu, program akan menyeleksi akar-akar yang menjadi dua akar riil (memotong sumbu-x), satu akar riil (menyinggung sumbu-x), dan tidak memiliki akar riil sama sekali (tidak memotong dan menyinggung). Dengan

1. if (discriminant > 0) : diskriminan merupakan bilangan positif dan menghasilkan nilai berupa dua akar riil.
2. else if (discriminant) : diskriminan bernilai nol dan menghasilkan hanya satu akar riil.
3. else : apabila diskriminan merupakan bilangan negatif (bukan positif dan nol) dan tidak menghasilkan akar sama sekali

Dengan demikian, nilai yang dihasilkan dari program adalah

```
Roots are complex and different.  
Root 1 = -1 + 1.41421i  
Root 2 = -1 - 1.41421i  
Testing complete! ° ◆ °
```

Number 3: Test Cases

Payslip Calculation Tests:

1. Test Case 1:

- **Employee Name:** Alice Brown
- **Base Salary:** Rp5,000,000
- **Percentage:** 100%
- **Installment:** Rp50,000
- **Insurance:** Rp250,000
- **Expected Net Salary:** Rp3,700,000

```
Name: Alice Brown  
Gross Salary: Rp5000000.00  
Tax (20%): Rp1000000.00  
Installment: Rp50000.00  
Insurance: Rp250000.00  
Net Salary: Rp3700000.00
```

Result: *The output matched the expected result. The program correctly handled the scenario with a lower base salary and higher insurance.*

Quadratic Equation Solver Tests:

1. Test Case 1 (Discriminant > 0):

- **Coefficients:** $a = 1, b = -3, c = 2$
- **Expected Roots:** 2 and 1 (two distinct real roots)

```
Testing Quadratic Equation Solver:  
Roots are real and different.  
Root 1 = 2.00  
Root 2 = 1.00
```

Result: *The output was correct, displaying two distinct real roots.*

2. Test Case 2 (Discriminant $= 0$):

- **Coefficients:** $a = 1, b = 2, c = 1$
- **Expected Root:** -1 (one real root)

```
Roots are real and the same.  
Root = -1.00
```

Result: *The output was correct, displaying one real root.*

3. Test Case 3 (Discriminant < 0):

- **Coefficients:** $a = 1, b = 2, c = 3$
- **Expected Roots:** Complex roots $(-1 \pm 1.41421i)$

```
Roots are complex and different.  
Root 1 = -1.00 + 1.41i  
Root 2 = -1.00 - 1.41i
```

Result: *The output was correct, displaying the complex roots accurately.*

Summary:

- *All test cases for the payslip calculation and quadratic equation solver passed successfully.*
- *The outputs matched the expected results in all scenarios.*
- *No discrepancies or issues were found during testing.*

Submission:

- *The test code file is provided above.*
- *The test report summarizes the test cases, results, and findings.*