# LAPORAN PRAKTIKUM

**ALGORITMA DAN PEMROGRAMAN**



Disusun Oleh:

WAFI FAHRUZZAMAN – 140810200009

PROGRAM STUDI S-1 TEKNIK INFORMATIKA FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS PADJADJARAN JATINANGOR

2020

## **Tujuan**

* 1. Memahami konsep *Record* dan *Array of Record* dalam pemrogram C++ secara benar.
  2. Mampu mengimplementasikan *Record* dan *Array of Record* pada program secara tepat.

## **Materi**

* 1. **Record**

Record merupakan suatu tipe data yang merupakan kumpulan dari atribut-atribut (field) suatu objek. Secara pendeklarasian, struct sangat berbeda dengan array yang hanya memiliki satu buah tipe data untuk setiap kumpulannya. Struct digunakan apabila data yang ingin dikelompokkan memiliki tipe data yang berbeda. Pendeklarasian struct sebagai berikut:

stuct data\_mahasiswa{

long int npm;

char nama[100];

char fakultas[100];

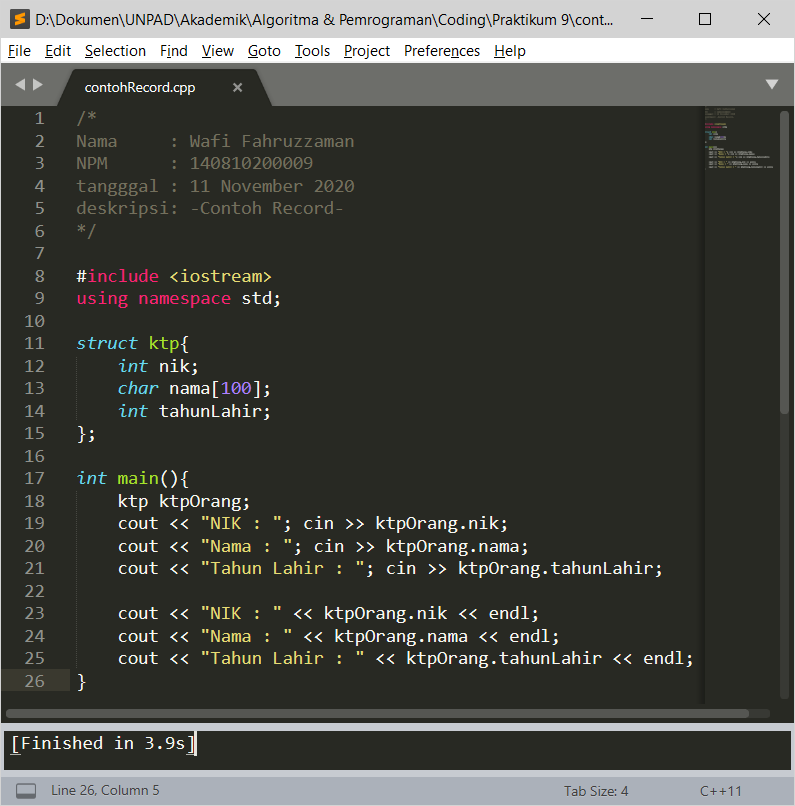
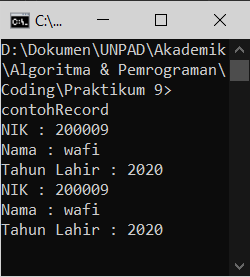
};

Untuk memanggil sebuah variabel yang berada didalam record, kita menggukan cara dengan bentuk dasar "namavar.namafield" contoh : mahasiswa.nama

* 1. **Array of Record**

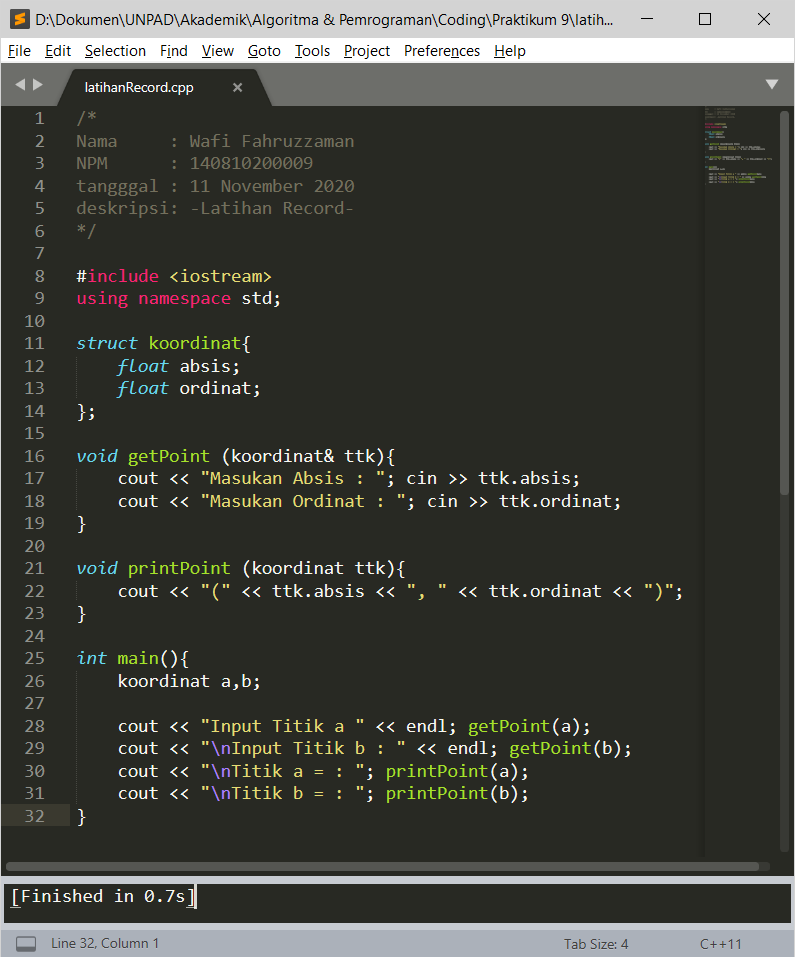
Salah satu penggunaan record adalah array of record atau mengakses record dengan array. Biasanya array of record dipakai untuk membuat program seperti dara mahasiswa, struk penjualan, dan perogram yang berhubungan dengan perekapan data lainnya.

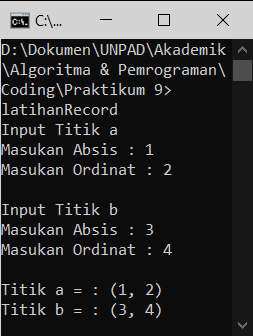
## **Latihan**

* 1. **Contoh Record**

Gambar 3.1.b. program/hasil (.exe)

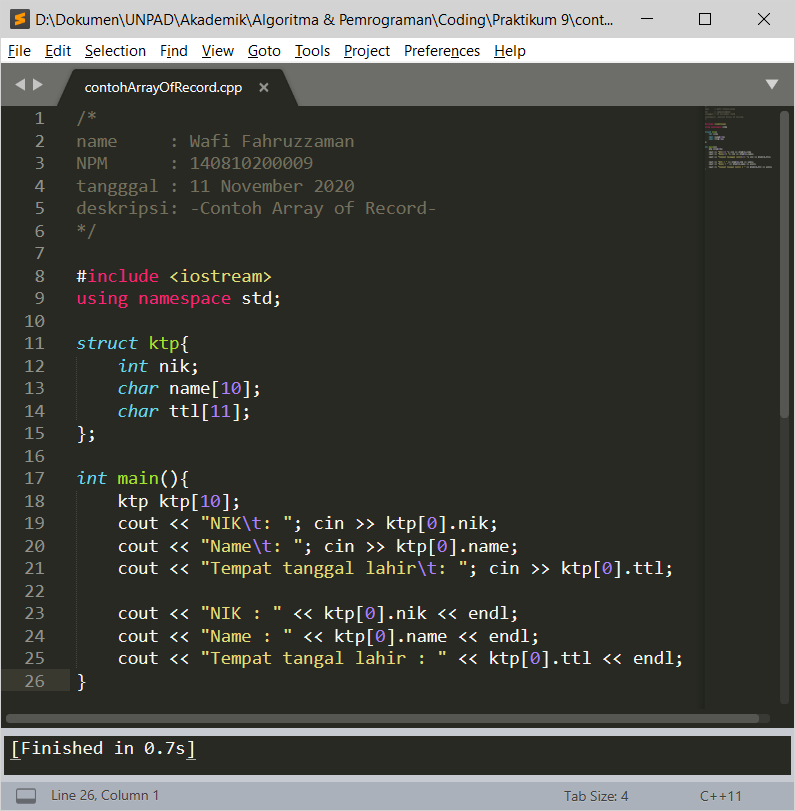
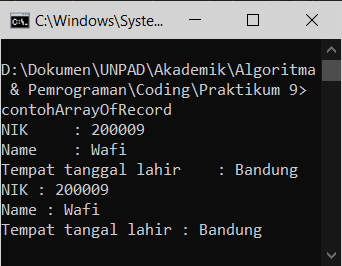
Gambar 3.1.a. coding (.cpp)

* 1. **Latihan Record**

****

*Gambar 3.2.a. coding (.cpp)*

*Gambar 3.2.b. program/hasil (.exe)*

* 1. **Contoh Array of Record**

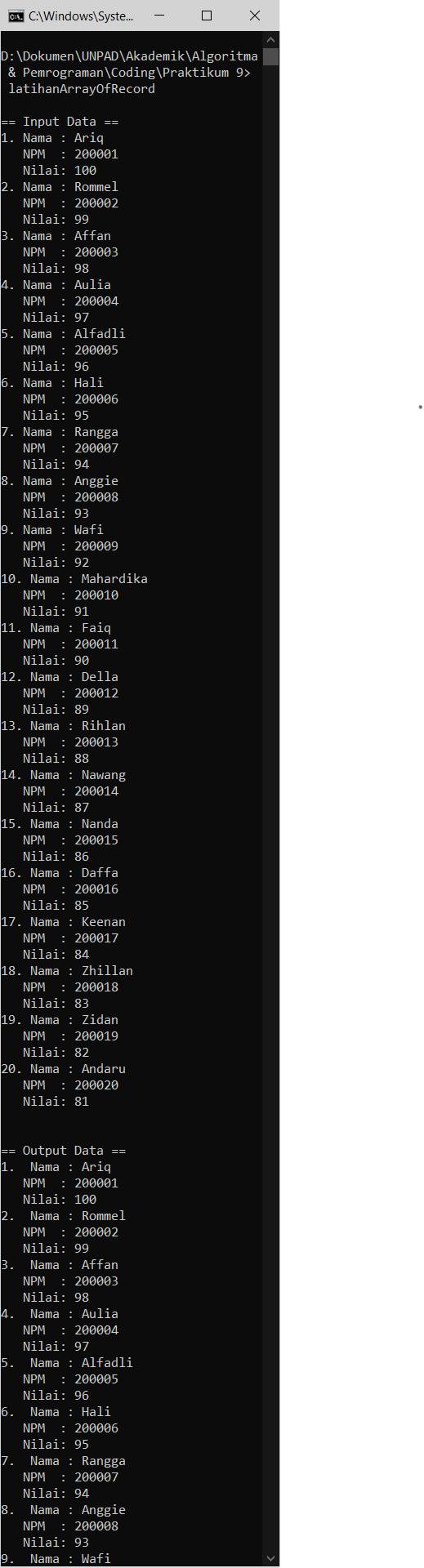
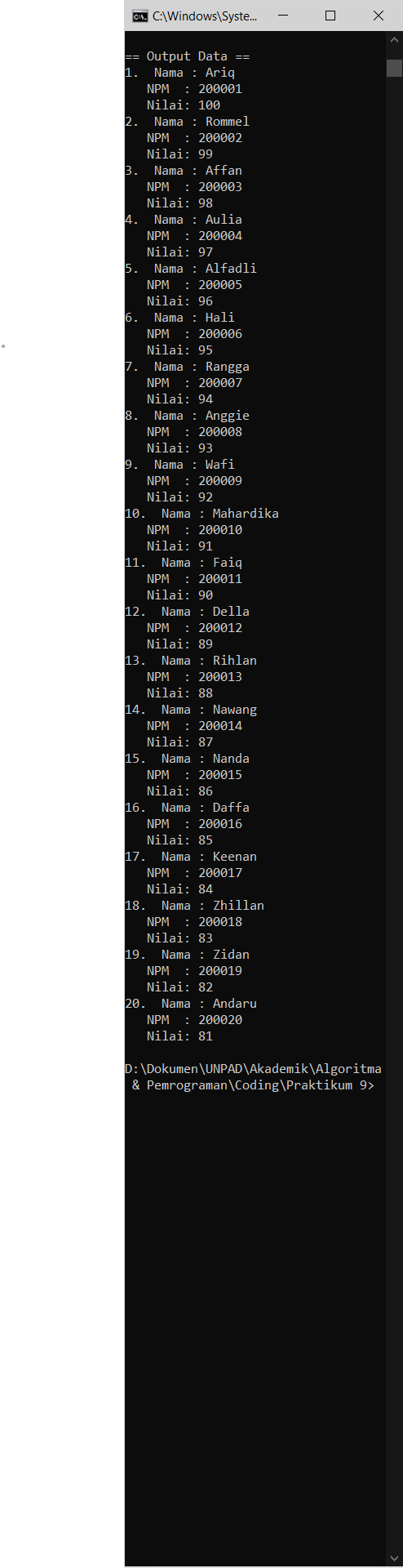
*Gambar 3.3.b. program/hasil (.exe)*

*Gambar 3.3.a. coding (.cpp)*

* 1. **Latihan Array of Record**

1. */\**
2. *Nama     : Wafi Fahruzzaman*
3. *NPM      : 140810200009*
4. *tangggal : 11 November 2020*
5. *deskripsi: -Latihan Array of Record-*
6. *\*/*
7. #include <iostream>
8. using namespace std;
9. struct mahasiswa{
10. char nama[15];
11. char npm[12];
12. int nilai;
13. };
14. int main(){
15. mahasiswa mahasiswa[20];
16. cout << "\n== Input Data ==\n";
17. for (int i = 0; i < 20; i++){
18. cout << i+1 << ". Nama : "; cin >> mahasiswa[i].nama;
19. cout << "   NPM  : "; cin >> mahasiswa[i].npm;
20. cout << "   Nilai: "; cin >> mahasiswa[i].nilai;
21. }

24. cout << "\n\n== Output Data ==\n";
25. for (int i = 0; i < 20; i++){
26. cout << i+1 << ".  Nama : " << mahasiswa[i].nama << endl;
27. cout << "   NPM  : " << mahasiswa[i].npm << endl;
28. cout << "   Nilai: " << mahasiswa[i].nilai << endl;
29. }
30. }

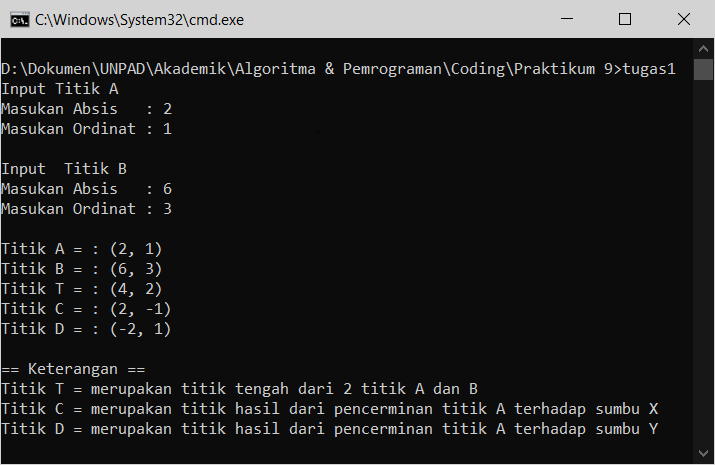
** Hasil Latihan Array of Record**

## **Tugas**

* 1. **Tugas 1**

1. */\**
2. *Nama     : Wafi Fahruzzaman*
3. *NPM      : 140810200009*
4. *tangggal : 11 November 2020*
5. *deskripsi: -Tugas 1-*
6. *\*/*
7. #include <iostream>
8. using namespace std;
9. struct koordinat{
10. float absis;
11. float ordinat;
12. };
13. void getPoint (koordinat& ttk);
14. void printPoint (koordinat ttk);
15. void keterangan ();
16. int main(){
17. koordinat a,b,c,d,t;
18. cout << "Input Titik A " << endl; getPoint(a);
19. cout << "\nInput  Titik B " << endl; getPoint(b);
20. cout << "\nTitik A = : "; printPoint(a);
21. cout << "\nTitik B = : "; printPoint(b);
22. t.absis = (a.absis + b.absis)/2;
23. t.ordinat = (a.ordinat + b.ordinat)/2;
24. cout << "\nTitik T = : "; printPoint(t);
25. c.absis = a.absis;
26. c.ordinat = -a.ordinat;
27. cout << "\nTitik C = : "; printPoint(c);
28. d.absis = -a.absis;
29. d.ordinat = a.ordinat;
30. cout << "\nTitik D = : "; printPoint(d);
31. keterangan();
32. }
33. void getPoint (koordinat& ttk){
34. cout << "Masukan Absis\t: "; cin >> ttk.absis;
35. cout << "Masukan Ordinat : "; cin >> ttk.ordinat;
36. }
37. void printPoint (koordinat ttk){
38. cout << "(" << ttk.absis << ", " << ttk.ordinat << ")";
39. }
40. void keterangan (){
41. cout << "\n\n== Keterangan ==\n";
42. cout << "Titik T = merupakan titik tengah dari 2 titik A dan B\n";
43. cout << "Titik C = merupakan titik hasil dari pencerminan titik A terhadap sumbu X\n";
44. cout << "Titik D = merupakan titik hasil dari pencerminan titik A terhadap sumbu Y\n ";
45. }

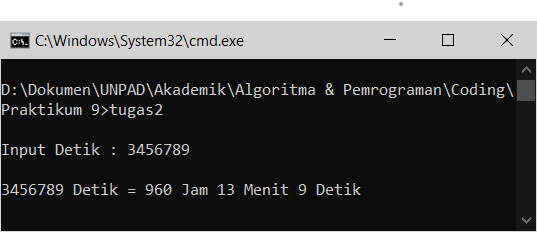
**Hasil Program Tugas 1 :**



* 1. **Tugas 2**

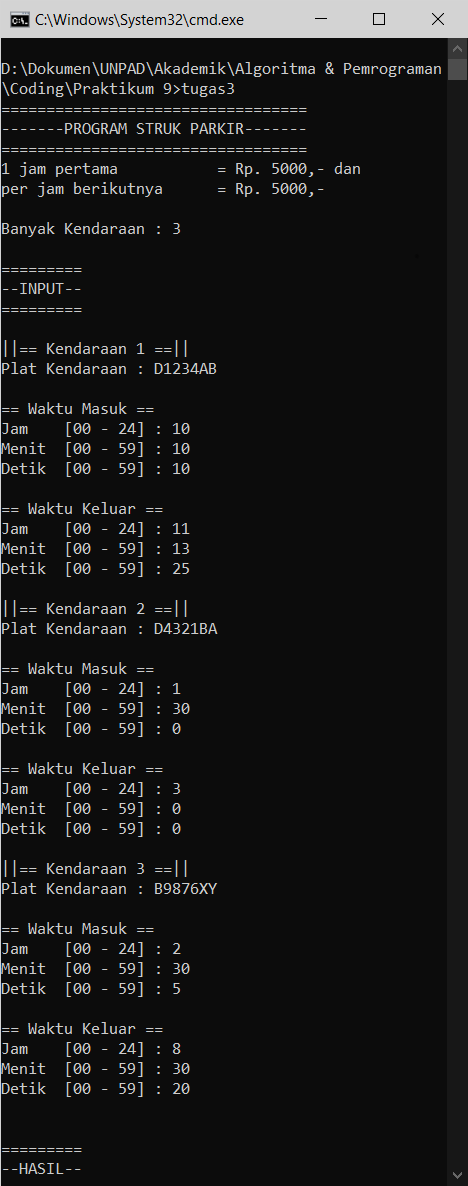
1. */\**
2. *Nama     : Wafi Fahruzzaman*
3. *NPM      : 140810200009*
4. *tangggal : 11 November 2020*
5. *deskripsi: -Tugas 2-*
6. *\*/*
7. #include <iostream>
8. using namespace std;
9. struct waktu{
10. int jam;
11. int menit;
12. int detik;
13. };
14. void inputDetik(waktu& input);
15. void proses(waktu& output, waktu input);
16. void print(waktu output, waktu input);
17. int main(){
18. waktu input, output;
19. inputDetik(input);
20. proses(output, input);
21. print(output, input);
22. cout << endl;
23. }
24. void inputDetik(waktu& input){
25. cout << "\nInput Detik : "; cin >> input.detik;
26. }
27. void proses(waktu& output, waktu input){
28. output.jam = input.detik/3600;
29. output.detik = input.detik%3600;
30. output.menit = output.detik/60;
31. output.detik = input.detik%60;
32. }
33. void print(waktu output, waktu input){
34. cout << endl;
35. cout << input.detik << " Detik = ";
36. cout << output.jam << " Jam ";
37. cout << output.menit << " Menit ";
38. cout << output.detik << " Detik " << endl;
39. }

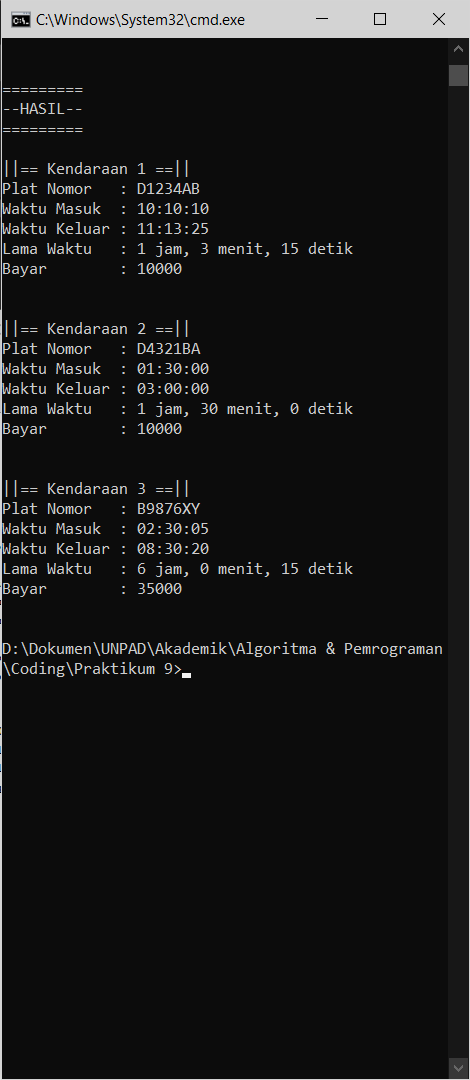
**Hasil Program Tugas 2 :**



* 1. **Tugas 3**

1. */\**
2. *Nama     : Wafi Fahruzzaman*
3. *NPM      : 140810200009*
4. *tangggal : 11 November 2020*
5. *deskripsi: -Tugas 3-*
6. *\*/*
7. #include <iostream>
8. #include <iomanip>
9. using namespace std;
10. struct parkir{
11. string plat;
12. int masuk;
13. int keluar;
14. int biaya;
15. };
16. int n;
17. parkir waktu[100];
18. int lamaWaktu[100], j[100], m[100], d[100] ;
19. void header(string isi);
20. int banyak();
21. void inputData(parkir kendaraan[], parkir jam[], parkir menit[], parkir detik[]);
22. void konversiWaktu(parkir jam[], parkir menit[], parkir detik[]);
23. void LamaWaktu(parkir jam[], parkir menit[], parkir detik[]);
24. void cariBiaya(parkir kendaraan[], int j[]);
25. void cariLamaWaktu(parkir waktu[]);
26. void konversiLamaWaktu(int lamaWaktu[]);
27. void cetakData(parkir kendaraan[], parkir jam[], parkir menit[], parkir detik[], int jj[], int mm[], int dd[]);
28. int main(){
29. parkir kendaraan[100];
30. parkir jam[100], menit[100], detik[100];
31. n = banyak();
32. inputData(kendaraan, jam, menit, detik);
33. konversiWaktu(jam, menit, detik);
34. cariLamaWaktu(waktu);
35. konversiLamaWaktu(lamaWaktu);
36. cariBiaya(kendaraan, j);
37. cetakData(kendaraan, jam, menit, detik, j, m, d);
38. }
39. void header(string isi){
40. for (int i=1; i<= isi.length(); i++){
41. cout << "=";
42. if (i == isi.length()){
43. cout << endl;
44. }
45. }
46. cout << isi << endl;
47. for (int i=1; i <= isi.length(); i++){
48. cout << "=";
49. if (i == isi.length()){
50. cout << endl;
51. }
52. }
53. }
54. int banyak(){
55. header("-------PROGRAM STRUK PARKIR-------");
56. cout << "1 jam pertama\t\t= Rp. 5000,- dan \n"
57. << "per jam berikutnya\t= Rp. 5000,-\n\n";
58. cout << "Banyak Kendaraan : "; cin >> n;
59. return n;
60. }
61. void inputData(parkir kendaraan[], parkir jam[], parkir menit[], parkir detik[]){
62. cout << endl;
63. header("--INPUT--");
64. for (int i = 0; i < n; i++){
65. cout << "\n||== Kendaraan " << i+1 << " ==||";
66. cout << "\nPlat Kendaraan : "; cin >> kendaraan[i].plat;
67. cout << "\n== Waktu Masuk ==\n";
68. cout << "Jam    [00 - 24] : "; cin >> jam[i].masuk;
69. cout << "Menit  [00 - 59] : "; cin >> menit[i].masuk;
70. cout << "Detik  [00 - 59] : "; cin >> detik[i].masuk;
71. cout << "\n== Waktu Keluar ==\n";
72. cout << "Jam    [00 - 24] : "; cin >> jam[i].keluar;
73. cout << "Menit  [00 - 59] : "; cin >> menit[i].keluar;
74. cout << "Detik  [00 - 59] : "; cin >> detik[i].keluar;
75. }
76. }
77. void konversiWaktu(parkir jam[], parkir menit[], parkir detik[]){
78. parkir jj[100], mm[100], dd[100];
79. for(int i; i<n; i++){
80. jj[i].masuk = jam[i].masuk \* 3600; mm[i].masuk = menit[i].masuk \* 60;
81. jj[i].keluar = jam[i].keluar \* 3600; mm[i].keluar = menit[i].keluar \* 60;
82. waktu[i].masuk = jj[i].masuk + mm[i].masuk + detik[i].masuk;
83. waktu[i].keluar = jj[i].keluar + mm[i].keluar + detik[i].keluar;
84. }
85. }
86. void cariLamaWaktu(parkir waktu[]){
87. for (int i = 0; i < n; i++){
88. lamaWaktu[i] = waktu[i].keluar - waktu[i].masuk;
89. }
90. }
91. void konversiLamaWaktu(int lamaWaktu[]){
92. for (int i = 0; i < n; i++){
93. if (lamaWaktu[i] < 0){
94. lamaWaktu[i] += 86400;
95. }
96. j[i] = lamaWaktu[i]/3600;
97. d[i] = lamaWaktu[i]%3600;
98. m[i] = d[i]/60;
99. d[i] = lamaWaktu[i]%60;
100. }
101. }
102. void cariBiaya(parkir kendaraan[], int j[]){
103. for (int i = 0; i < n; i++){
104. kendaraan[i].biaya = (j[i]+1) \* 5000;
105. }
106. }
107. void cetakData(parkir kendaraan[], parkir jam[], parkir menit[], parkir detik[], int jj[], int mm[], int dd[]){
108. cout << endl << endl;;
109. header("--HASIL--");
110. for (int i = 0; i < n; i++){
111. if ((jam[i].masuk >= 01 && jam[i].masuk <=24) && (menit[i].masuk >= 00 && menit[i].masuk <= 59) && (detik[i].masuk >= 00 && detik[i].masuk <=59)){
112. if ((jam[i].keluar >= 01 && jam[i].keluar <=24) && (menit[i].keluar >= 00 && menit[i].keluar <= 59) && (detik[i].keluar >= 00 && detik[i].keluar <=59)){
113. cout << "\n||== Kendaraan " << i+1 << " ==||";
114. cout << "\nPlat Nomor   : " << kendaraan[i].plat << endl;
115. cout << "Waktu Masuk  : "
116. << setfill('0') << setw(2) << jam[i].masuk << ":"
117. << setfill('0') << setw(2) << menit[i].masuk << ":"
118. << setfill('0') << setw(2) << detik[i].masuk << endl;
119. cout << "Waktu Keluar : "
120. << setfill('0') << setw(2) << jam[i].keluar << ":"
121. << setfill('0') << setw(2) << menit[i].keluar << ":"
122. << setfill('0') << setw(2) << detik[i].keluar << endl;
123. cout << "Lama Waktu   : "
124. << j[i] << " jam, "
125. << m[i] << " menit, "
126. << d[i] << " detik" << endl;
127. cout << "Bayar        : "
128. << kendaraan[i].biaya << endl << endl;
129. } else{
130. cout << "\nMasukkan kembali input waktu keluar pada Data " << i+1 << " dengan benar!" << endl;
131. }
132. } else {
133. cout << "\nMasukkan kembali input Data " << i+1 << " dengan benar!" << endl;
134. }
135. }
136. }

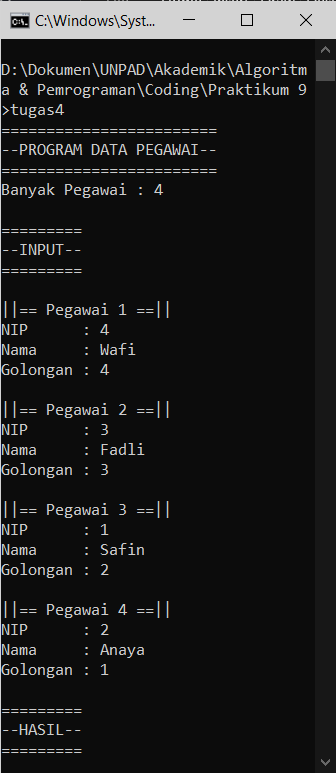
** Hasil Tugas 3**

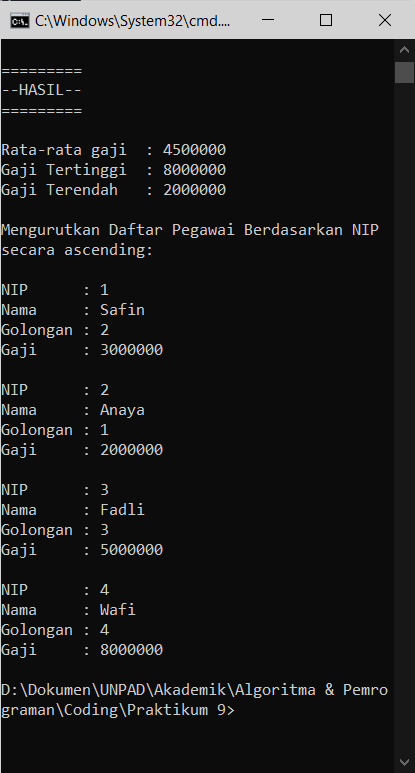
****

* 1. **Tugas 4**

1. */\**
2. *Nama     : Wafi Fahruzzaman*
3. *NPM      : 140810200009*
4. *tangggal : 11 November 2020*
5. *deskripsi: -Tugas 4-*
6. *\*/*
7. #include <iostream>
8. using namespace std;
9. int n;
10. struct pegawai{
11. string nip, nama;
12. int gol;
13. long gaji;
14. };
15. void header(string isi);
16. int banyakPegawai();
17. void inputPegawai(pegawai data[]);
18. void outputPegawai(pegawai data[]);
19. void gajiPegawai(pegawai data[]);
20. long rataanGaji(pegawai data[]);
21. long gajiTertinggi(pegawai data[]);
22. long gajiTerendah(pegawai data[]);
23. void swap (int &a, int&b);
24. void mengurutkanNIP(pegawai data[]);
25. int main(){
26. pegawai data[1000];
27. int banyak;
28. long rata2, tertinggi, terendah;
29. banyak = banyakPegawai();
30. inputPegawai(data);
31. gajiPegawai(data);
32. rata2 = rataanGaji(data);
33. tertinggi = gajiTertinggi(data);
34. terendah = gajiTerendah(data);
35. mengurutkanNIP(data);
36. cout << endl;
37. header("--HASIL--");
38. cout << "\nRata-rata gaji  : " << rata2
39. << "\nGaji Tertinggi  : " << tertinggi
40. << "\nGaji Terendah   : " << terendah
41. << endl;
42. outputPegawai(data);
43. }
44. void header(string isi){
45. for (int i=1; i<= isi.length(); i++){
46. cout << "=";
47. if (i == isi.length()){
48. cout << endl;
49. }
50. }
51. cout << isi << endl;
52. for (int i=1; i <= isi.length(); i++){
53. cout << "=";
54. if (i == isi.length()){
55. cout << endl;
56. }
57. }
58. }
59. int banyakPegawai(){
60. header("--PROGRAM DATA PEGAWAI--");
61. cout << "Banyak Pegawai : "; cin >> n;
62. return n;
63. }
64. void inputPegawai(pegawai data[]){
65. cout << endl;
66. header("--INPUT--");
67. for (int i = 0; i < n; i++){
68. cout << "\n||== Pegawai " << i+1 << " ==||";
69. cout << "\nNIP      : "; cin >> data[i].nip;
70. cout << "Nama     : "; cin >> data[i].nama;
71. cout << "Golongan : "; cin >> data[i].gol;
72. }
73. }
74. void outputPegawai(pegawai data[]){
75. cout << "\nMengurutkan Daftar Pegawai Berdasarkan NIP "
76. << "secara ascending:\n";
77. for (int i = 0; i < n; i++){
78. cout << "\nNIP      : " << data[i].nip
79. << "\nNama     : " << data[i].nama
80. << "\nGolongan : " << data[i].gol
81. << "\nGaji     : " << data[i].gaji << endl;
82. }
83. }
84. void gajiPegawai(pegawai data[]){
85. for (int i = 0; i < n; i++){
86. switch(data[i].gol){
87. case 1 : data[i].gaji = 2000000; break;
88. case 2 : data[i].gaji = 3000000; break;
89. case 3 : data[i].gaji = 5000000; break;
90. case 4 : data[i].gaji = 8000000; break;
91. }
92. }
93. }
94. long rataanGaji(pegawai data[]){
95. long rata2, jumlah=0;
96. for (int i = 0; i < n; i++){
97. jumlah += data[i].gaji ;
98. }
99. rata2 = jumlah / n;
100. return rata2;
101. }
102. long gajiTertinggi(pegawai data[]){
103. long tertinggi = 0;
104. for (int i = 0; i < n; i++) {
105. if(data[i].gaji > tertinggi){
106. tertinggi = data[i].gaji;
107. }
108. }
109. return tertinggi;
110. }
111. long gajiTerendah(pegawai data[]){
112. long terendah = 8000000;
113. for (int i = 0; i < n; i++) {
114. if(data[i].gaji < terendah){
115. terendah = data[i].gaji;
116. }
117. }
118. return terendah;
119. }
120. void swap (int &a, int&b){
121. int temp = a;
122. a = b;
123. b = temp;
124. }
125. void mengurutkanNIP(pegawai data[]){
126. for (int i = n-1; i > 0; i--){
127. for (int j = 0; j < i; j++){
128. if (data[j].nip > data[j+1].nip)
129. swap(data[j], data[j+1]);
130. }
131. }
132. }

**Hasil Tugas 4**

****

****

1. **Kesimpulan**

Pada praktikum Algoritma dan Pemrograman ke 9, telah mempelajari dan mengimplementasi-kan materi yang diberikan yaitu

1. Record
2. Array of Record

Selain mempelajari materi-materi tersebut, juga telah mengimplementasikan dengan cara membuat program sederhana. Pada praktikum ini membuat program sederhana yaitu membuat Program Koordinat. Program tersebut dapat mencari titik tengah dari titik A dan titik B. Juga, dapat mencerminkan titik A terhadap sumbu X. Juga, dapat mencerminkan titik A terhadap sumbu Y.

Setelah itu, membuat program Konversi Waktu. User meng-input detik kemudian nanti akan ditampilkan berupa jam, menit, detik.

Kemudian, membuat program Struk Parkir. User meng-input banyak kendaraan, input plat nomor, waktu masuk, dan waktu keluar kendaraan. Kenudian, akan ditampilkan plat nomor, waktu masuk, waktu keluar, lama waktu parkir, dan harga yang harus dibayar.

Selain itu, membuat program Program Data Pegawai. User meng-input banyak pegawai, input NIP, nama, dan golongan pegawai. Kemudian, akan ditampilkan rata- rata gaji, gaji tertinggi, gaji terendah, kemudian daftar pegawai yang telah diurutkan berdasarkan NIP.