**LAPORAN PRAKTIKUM**

**PEMROGRAMAN I**

**MODUL 6**

****

**ARRAY**

**Oleh:**

**M. Anshary NIM. 2410817310008**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**DESEMBER 2024**

# **LEMBAR PENGESAHAN**

**LAPORAN PRAKTIKUM PEMROGRAMAN I**

**MODUL 6**

Laporan Praktikum Pemrograman I Modul 6 : Array ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman I. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : M. Anshary

NIM : 2410817310008

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Muhammad Ryan Rizky Rahmadi  NIM. 2210817310001 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Helda Yunita, S.Kom., M.Kom.  NIP. 199106192024062001 |

# **DAFTAR ISI**

[**LEMBAR PENGESAHAN** 2](#_Toc185627110)

[**DAFTAR ISI** 3](#_Toc185627111)

[**DAFTAR TABEL** 4](#_Toc185627112)

[**DAFTAR GAMBAR** 5](#_Toc185627113)

[**SOAL 1** 6](#_Toc185627114)

[**A.** **Source Code** 7](#_Toc185627115)

[**B.** **Output Program** 8](#_Toc185627116)

[**C.** **Pembahasan** 8](#_Toc185627117)

[**SOAL 2** 10](#_Toc185627118)

[**A.** **Source Code** 11](#_Toc185627119)

[**B.** **Output Program** 11](#_Toc185627120)

[**C.** **Pembahasan** 12](#_Toc185627121)

[**SOAL 3** 14](#_Toc185627122)

[**A.** **Source Code** 15](#_Toc185627123)

[**B.** **Output Program** 16](#_Toc185627124)

[**C.** **Pembahasan** 16](#_Toc185627125)

[**SOAL 4** 19](#_Toc185627126)

[**A.** **Source Code** 21](#_Toc185627127)

[**B.** **Output Program** 23](#_Toc185627128)

[**C.** **Pembahasan** 23](#_Toc185627129)

[**SOAL 5** 26](#_Toc185627130)

[**A.** **Source Code** 27](#_Toc185627131)

[**B.** **Output Program** 28](#_Toc185627132)

[**C.** **Pembahasan** 29](#_Toc185627133)

# **DAFTAR TABEL**

[Tabel 1. 1 Source Code Soal 1 Bahasa C 7](#_Toc185627090)

[Tabel 1. 2 Source Code Soal 1 Bahasa Python 7](#_Toc185627091)

[Tabel 1. 3 Source Code Soal 2 Bahasa C 11](#_Toc185627092)

[Tabel 1. 4 Source Code Soal 2 Bahasa Python 11](#_Toc185627093)

[Tabel 1. 5 Source Code Soal 3 Bahasa C 15](#_Toc185627094)

[Tabel 1. 6 Source Code Soal 3 Bahasa Python 15](#_Toc185627095)

[Tabel 1. 7 Source Code Soal 4 Bahasa C 21](#_Toc185627096)

[Tabel 1. 8 Source Code Soal 4 Bahasa Python 22](#_Toc185627097)

[Tabel 1. 9 Source Code Soal 5 Bahasa C 27](#_Toc185627098)

[Tabel 1. 10 Source Code Soal 5 Bahasa Python 28](#_Toc185627099)

# **DAFTAR GAMBAR**

[Gambar 1. 1 Screenshot Output Soal 1 Bahasa C 8](#_Toc185627100)

[Gambar 1. 2 Screenshot Output Soal 1 Bahasa Python 8](#_Toc185627101)

[Gambar 1. 3 Screenshot Output Soal 2 Bahasa C 11](#_Toc185627102)

[Gambar 1. 4 Screenshot Output Soal 2 Bahasa Python 12](#_Toc185627103)

[Gambar 1. 5 Screenshot Output Soal 3 Bahasa C 16](#_Toc185627104)

[Gambar 1. 6 Screenshot Output Soal 3 Bahasa Python 16](#_Toc185627105)

[Gambar 1. 7 Screenshot Output Soal 4 Bahasa C 23](#_Toc185627106)

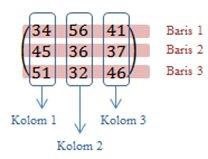
[Gambar 1. 8 Screenshot Output Soal 4 Bahasa Python 23](#_Toc185627107)

[Gambar 1. 9 Screenshot Output Soal 5 Bahasa C 28](#_Toc185627108)

[Gambar 1. 10 Screenshot Output Soal 5 Bahasa Python 29](#_Toc185627109)

# **SOAL 1**

1. Matriks adalah kumpulan bilangan yang disusun secara baris dan kolom yang kemudian diisi dengan angka-angka pada matriks tersebut. Misalnya sebuah matriks memiliki 3 baris dan 3 kolom dengan isi yang ada di dalam matriks tersebut adalah {(34), (56), (41), (45), (36), (37), (51), (32), (46)} maka akan terbentuk matriks sebagai berikut:



Untuk mendapatkan kekuatan *One For All* Midoriya Izuku harus membuat sebuah matriks sesuai dengan baris dan kolom yang ditetapkan beserta isi yang ada didalamnya. Buatlah sebuah program untuk membantu Midoriya Izuku membuat matriks.

Format input:

Input pada baris pertama berupa jumlah baris dan kolom.

Input pada baris kedua berupa angka yang ada di dalam matriks tersebut.

Jumlah input pada baris kedua = (baris x kolom) yang dipisahkan dengan spasi.

Format output:

Output merupakan sebuah matriks .

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2 3  1 2 3 4 5 6 | 1 2 3  4 5 6 |
| 3 3  34 56 41 45 36 37 51 32 46 | 34 56 41 45 36 37  51 32 46 |
| 4 5  1 1 1 1 2 3 5 6 4 5 8 7 9 6 5 4 1 2 5 6 | 1 1 1 1 2 3 5 6 4 5  8 7 9 6 5  4 1 2 5 6 |

Simpan coding anda dengan nama: **PRAK601-NIM-Nama.py** dan **PRAK601-NIM-Nama.c**

## **Source Code**

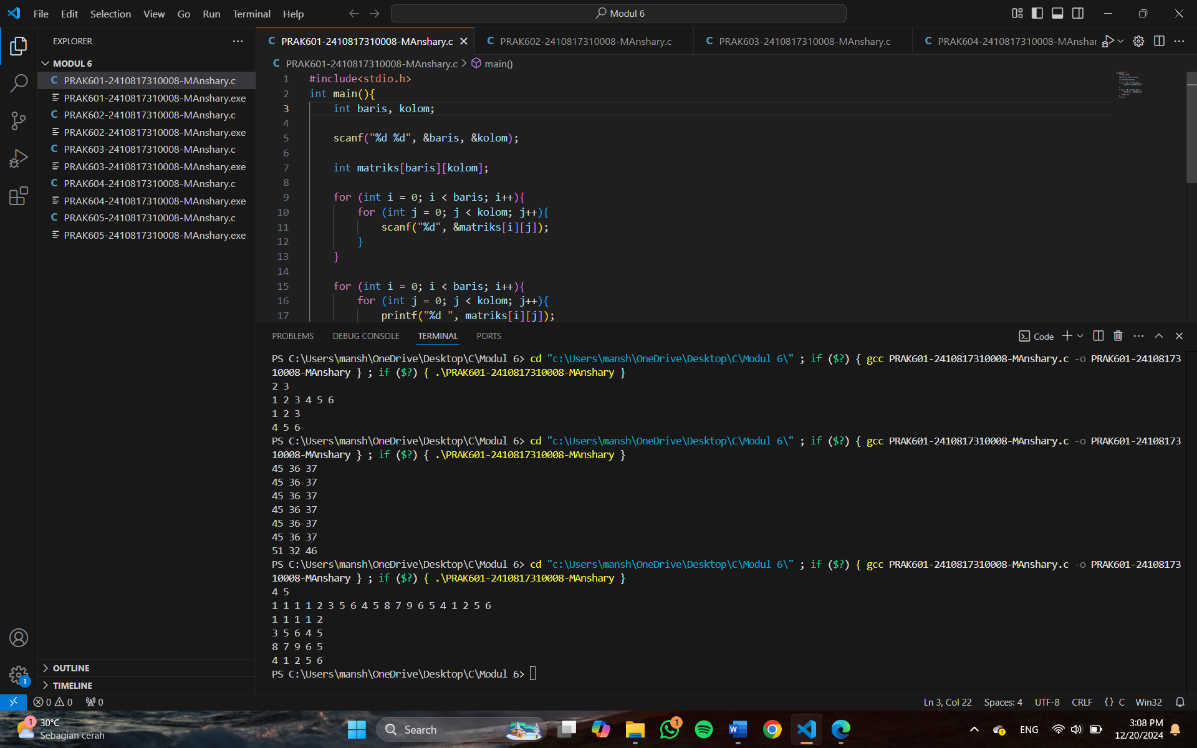
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 18 19 20  21  22 | #include<stdio.h>  int main(){      int baris, kolom;        scanf("%d %d", &baris, &kolom);        int matriks[baris][kolom];        for (int i = 0; i < baris; i++){          for (int j = 0; j < kolom; j++){              scanf("%d", &matriks[i][j]);          }      }      for (int i = 0; i < baris; i++){          for (int j = 0; j < kolom; j++){              printf("%d ", matriks[i][j]);          }          printf("\n");      }      return 0;  } |

Tabel 1. 1 Source Code Soal 1 Bahasa C

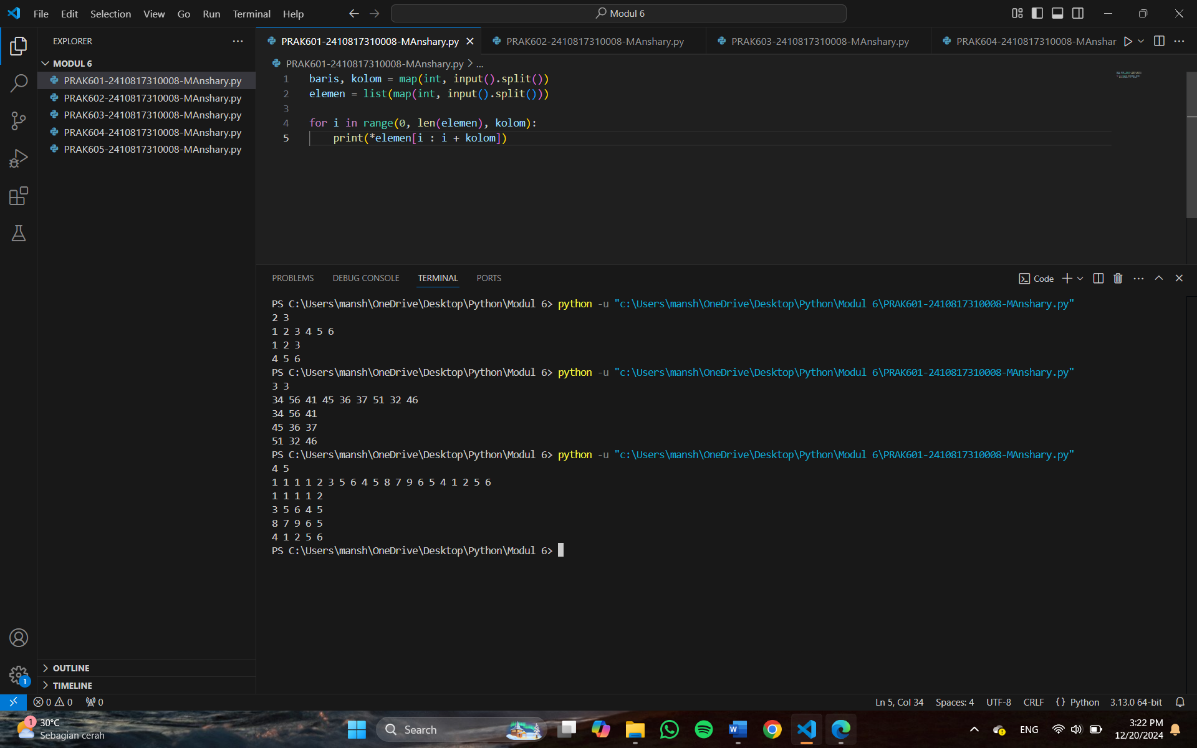
|  |  |
| --- | --- |
| 1  2  3  4  5 | baris, kolom = map(int, input().split())  elemen = list(map(int, input().split()))  for i in range(0, len(elemen), kolom):      print(\*elemen[i : i + kolom]) |

Tabel 1. 2 Source Code Soal 1 Bahasa Python

## **Output Program**



Gambar 1. 1 Screenshot Output Soal 1 Bahasa C



Gambar 1. 2 Screenshot Output Soal 1 Bahasa Python

## **Pembahasan**

1. Bahasa C :
2. **#include<stdio.h>** yang berfungsi untuk menyertakan file header ke dalam program. Salah satu contoh fungsi dari **#include<stdio.h>** yaitu **printf** dan **scanf**.
3. **int main** yang berfungsi sebagai titik awal program C.
4. **int baris** berfungsi untuk menyimpan variabel yang bersifat desimal (integer).
5. **scanf** digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel yang ditentukan.
6. **printf** yang berfungsi menampilkan output yang telah diketikkan.
7. **%d** : untuk membaca desimal (integer) yang dimasukkan oleh pengguna.
8. **&baris** : berfungsi menyimpan tipe data variabel pada poin **int.**
9. Lalu pada setiap akhir pernyataan ditambahkan **;** untuk menandakan bahwa pernyataan tersebut telah selesai.
10. **matriks[...][...]** : Menentukan ukuran dari array **dua** dimensi.
11. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
12. **\n** : bermakna newline (baris baru).
13. Terdapat Arithmetic Operator berupa++.
14. Terdapat juga Comparison Operators berupa : <.
15. **return 0** yang berfungsi mengembalikan program kembali seperti semula setelah dieksekusi.
16. Bahasa Python :
17. **baris, kolom** berfungsi untuk menyimpan variabel.
18. **list** : Mengubah objek **map** menjadi **list** yang sebenarnya, sehingga kita bisa bekerja dengan elemen-elemen tersebut seperti dengan **list** pada umumnya.
19. **map** :untuk melakukan operasi pada banyak elemen sekaligus.
20. **int** :digunakan untuk mengonversi tipe data menjadi tipe **integer** (bilangan desimal).
21. **input**() : digunakan untuk menerima input dari pengguna, lalu mengembalikan nilai yang dimasukkan sebagai string.
22. **split()** : untuk memisahkan string input yang dihasilkan oleh input() menjadi beberapa bagian berdasarkan spasi.
23. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
24. **range** : Menghasilkan urutan angka di dalam tanda ().
25. **len** : Menghitung jumlah elemen dalam list elemen.
26. **\*elemen** : fungsi \* tersebut adalah untuk membongkar (unpack) elemen-elemen dalam slice tersebut, sehingga elemen-elemen tersebut dicetak terpisah oleh spasi.
27. Terdapat Arithmetic Operator berupa+.
28. **print** yang berfungsi menampilkan output yang telah diketikkan.

# **SOAL 2**

1. Jarak Zetsu putih merupakan karakter yang mampu membelah diri dengan kelipatan sesuai dengan label ruangan yang ada.

Misalnya terdapat 3 ruangan dengan rincian sebagai berikut:

Pada ruangan berlabel 1 dimasukkan 4 zetsu putih, pada ruangan berlabel 2 dimasukkan 7 zetsu putih, pada ruangan berlabel 3 dimasukkan 9 zetsu putih. Setelah beberapa saat, zetsu putih yang ada pada ruangan berlabel 1 menjadi 4×1 zetsu putih, zetsu putih yang ada pada ruangan berlabel 2 menjadi 7×2 zetsu putih, zetsu putih yang ada pada ruangan berlabel 3 menjadi 9×3 zetsu putih. Buatlah sebuah program untuk menghitung jumlah zetsu putih pada setiap ruangan.

Format input:

Baris pertama merupakan jumlah ruangan.

Baris kedua merupakan banyaknya zetsu putih yang dimasukkan pada setiap ruangan.

Format output:

Output merupakan jumlah zetsu putih setelah membelah diri pada setiap ruangan.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  4 7 9 | 4 14 27 |
| 5  1 2 3 4 5 | 1 4 9 16 25 |
| 10  5 6 45 78 21 3 6 8 45 1 | 5 12 135 312 105 18 42 64 405 10 |

Simpan coding anda dengan nama: **PRAK602-NIM-Nama.py** dan **PRAK602-NIM-Nama.c**

## **Source Code**

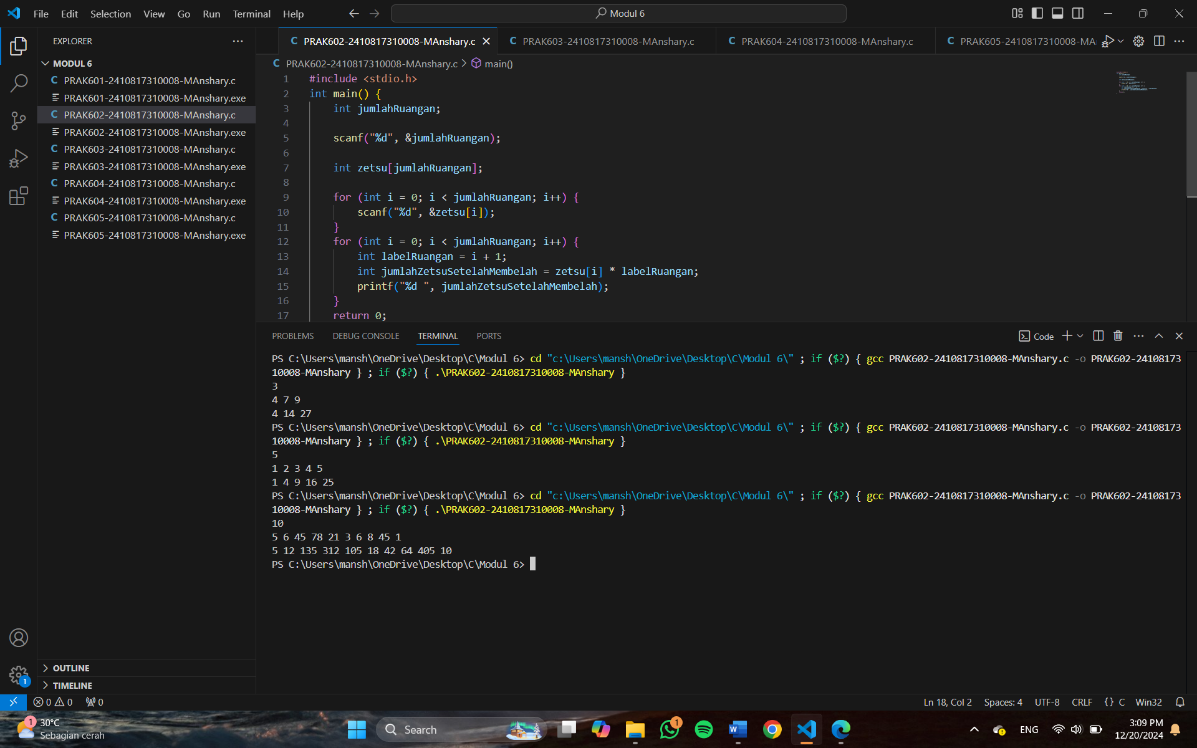
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | #include <stdio.h>  int main() {      int jumlahRuangan;      scanf("%d", &jumlahRuangan);      int zetsu[jumlahRuangan];      for (int i = 0; i < jumlahRuangan; i++) {          scanf("%d", &zetsu[i]);      }      for (int i = 0; i < jumlahRuangan; i++) {          int labelRuangan = i + 1;          int jumlahZetsuSetelahMembelah = zetsu[i] \* labelRuangan;          printf("%d ", jumlahZetsuSetelahMembelah);      }      return 0;  } |

Tabel 1. 3 Source Code Soal 2 Bahasa C

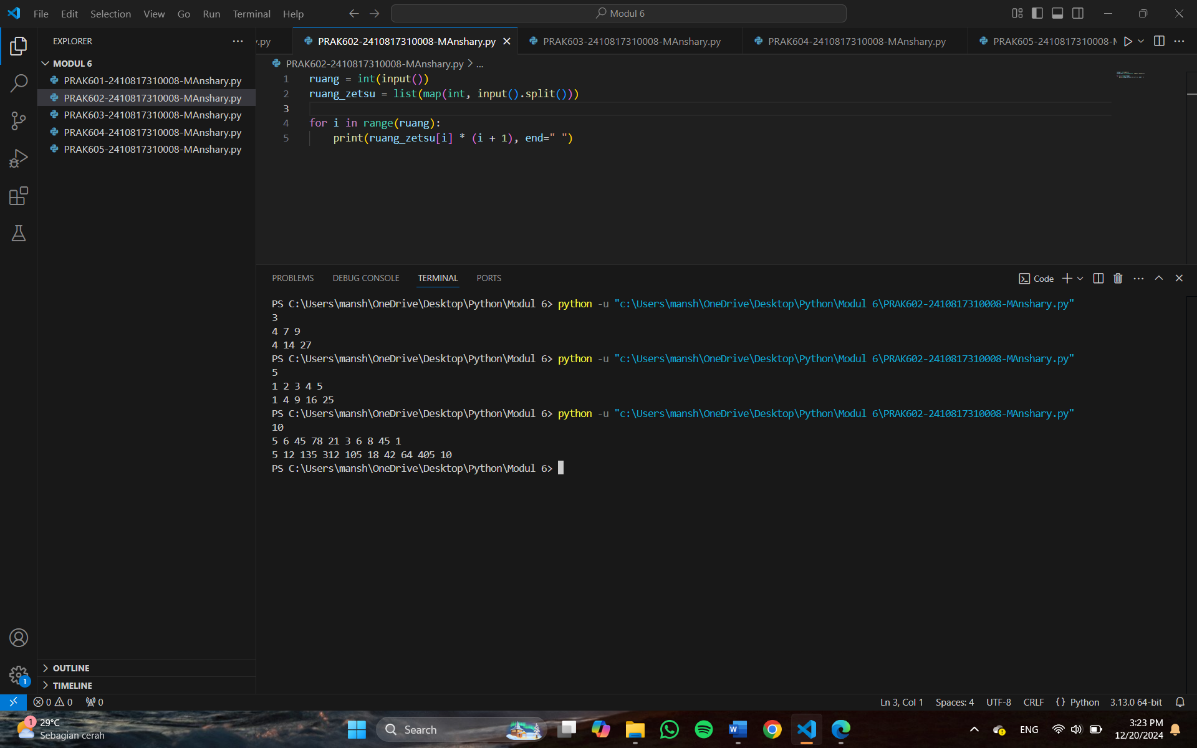
|  |  |
| --- | --- |
| 1  2  3  4  5 | ruang = int(input())  ruang\_zetsu = list(map(int, input().split()))  for i in range(ruang):      print(ruang\_zetsu[i] \* (i + 1), end=" ") |

Tabel 1. 4 Source Code Soal 2 Bahasa Python

## **Output Program**



Gambar 1. 3 Screenshot Output Soal 2 Bahasa C



Gambar 1. 4 Screenshot Output Soal 2 Bahasa Python

## **Pembahasan**

1. Bahasa C :
2. **#include<stdio.h>** yang berfungsi untuk menyertakan file header ke dalam program. Salah satu contoh fungsi dari **#include<stdio.h>** yaitu **printf** dan **scanf**.
3. **int main** yang berfungsi sebagai titik awal program C.
4. **int jumlahRuangan** digunakan untuk menyimpan data variabel **integer** (bilangan desimal).
5. **scanf** digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel yang ditentukan.
6. **printf** yang berfungsi menampilkan output yang telah diketikkan.
7. **%d** : untuk membaca desimal (integer) yang dimasukkan oleh pengguna.
8. **&jumlahRuangan** : berfungsi menyimpan tipe data variabel pada poin **int.**
9. Lalu pada setiap akhir pernyataan ditambahkan **;** untuk menandakan bahwa pernyataan tersebut telah selesai.
10. **zetsu[...]** : Menentukan ukuran dari array **satu** dimensi.
11. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
12. Terdapat Arithmetic Operator berupa++, +, \*.
13. Terdapat juga Comparison Operators berupa : <.
14. **return 0** yang berfungsi mengembalikan program kembali seperti semula setelah dieksekusi.
15. Bahasa Python :
16. **ruang** berfungsi untuk menyimpan variabel.
17. **list** : Mengubah objek **map** menjadi **list** yang sebenarnya, sehingga kita bisa bekerja dengan elemen-elemen tersebut seperti dengan **list** pada umumnya.
18. **map** :untuk melakukan operasi pada banyak elemen sekaligus.
19. **int** :digunakan untuk mengonversi tipe data lain menjadi tipe **integer** (bilangan desimal).
20. **input**() : digunakan untuk menerima input dari pengguna, lalu mengembalikan nilai yang dimasukkan sebagai string.
21. **split()** : untuk memisahkan string input yang dihasilkan oleh input() menjadi beberapa bagian berdasarkan spasi.
22. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
23. **range** : Menghasilkan urutan angka di dalam tanda ().
24. **ruang\_zetsu[...]** : Menentukan ukuran dari array **satu** dimensi.
25. **print** yang berfungsi menampilkan output yang telah diketikkan.
26. **end=” ”** : untuk mengontrol apa yang dicetak setelah setiap output, disini berfungsi untuk membuat output agar tidak menjalankan newline pada setiap loop.
27. Terdapat Arithmetic Operator berupa\*, **+**.

# **SOAL 3**

1. Pak Ruli adalah seorang guru tik dan matematika di sebuah smp rumah bangsa. Ruli ingin membuat perkalian bilangan antara baris 1 dan baris 2.

Misal di beri angka n1=2 dan n2=2 maka inputan baris selanjutnya 2 kali untuk baris pertama dan 2 kali juga untuk baris kedua. Lalu angka untuk baris pertama 2 dan 3 selanjutnya angka untuk baris kedua 4 dan 5. Jadi perkaliannya itu baris1 kolom1 dikali baris2 kolom1 selanjutnya baris1 kolom2 dikali baris2 kolom2. Berikut rumusnya :

|  |
| --- |
| 2 3  4 5  Jadi hasilnya itu 8 15 |

Format input :

bilangan n1 dan n2; jika n1 dan n2 tidak sama maka menampilkan jumlah tidak sama. jika n1 dan n2 sama maka inputan angka sebanyak 2\*n1 atau 2\*n2.

Format Output :

hasil output adalah baris pertama di kali baris ke dua, jadi liat contoh input output untuk lebih jelas

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 3  1 2 3  4 5 6 | 4 10 18 |
| 2 3 | Jumlah tidak sama |
| 5 5  1 2 3 4 5  5 4 3 2 1 | 5 8 9 8 5 |

Simpan coding anda dengan nama: **PRAK603-NIM-Nama.py** dan **PRAK603-NIM-Nama.c**

## **Source Code**

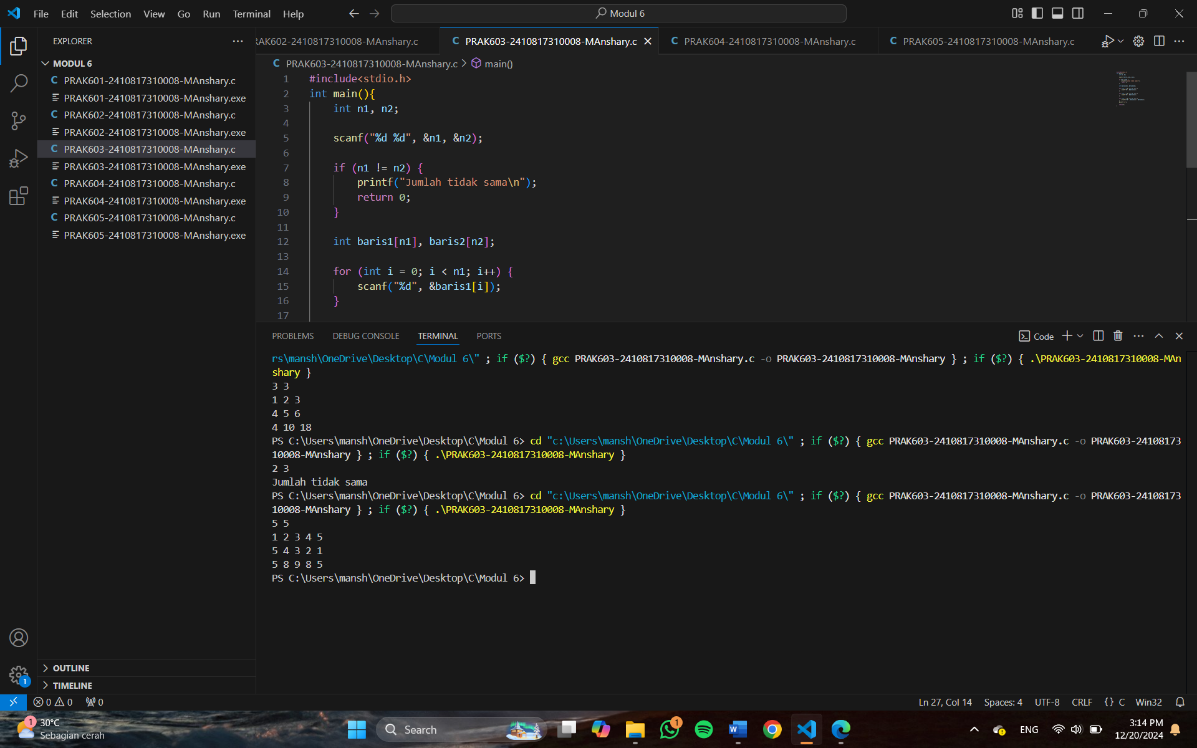
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | #include<stdio.h>  int main(){      int n1, n2;      scanf("%d %d", &n1, &n2);      if (n1 != n2) {          printf("Jumlah tidak sama\n");          return 0;      }      int baris1[n1], baris2[n2];      for (int i = 0; i < n1; i++) {          scanf("%d", &baris1[i]);      }      for (int i = 0; i < n2; i++) {          scanf("%d", &baris2[i]);      }      for (int i = 0; i < n1; i++) {          printf("%d ", baris1[i] \* baris2[i]);      }      printf("\n");      return 0;  } |

Tabel 1. 5 Source Code Soal 3 Bahasa C

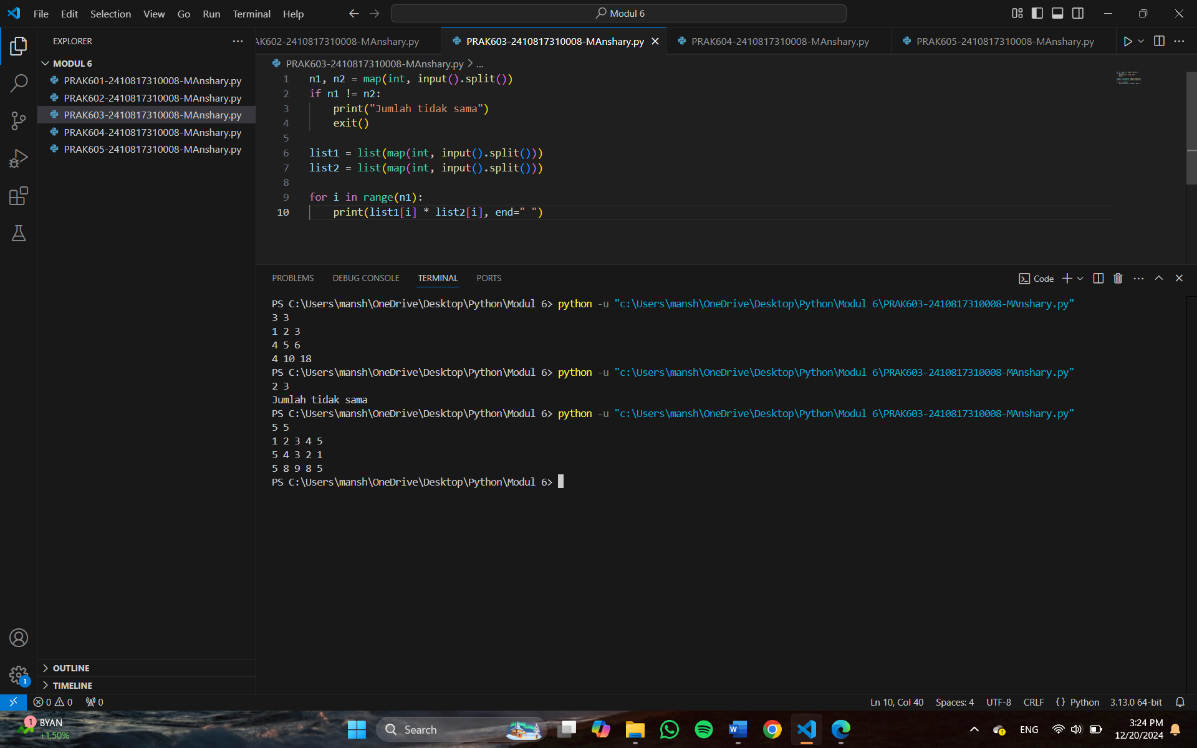
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | n1, n2 = map(int, input().split())  if n1 != n2:      print("Jumlah tidak sama")      exit()    list1 = list(map(int, input().split()))  list2 = list(map(int, input().split()))  for i in range(n1):      print(list1[i] \* list2[i], end=" ") |

Tabel 1. 6 Source Code Soal 3 Bahasa Python

## **Output Program**



Gambar 1. 5 Screenshot Output Soal 3 Bahasa C



Gambar 1. 6 Screenshot Output Soal 3 Bahasa Python

## **Pembahasan**

1. Bahasa C :
2. **#include<stdio.h>** yang berfungsi untuk menyertakan file header ke dalam program. Salah satu contoh fungsi dari **#include<stdio.h>** yaitu **printf** dan **scanf**.
3. **int main** yang berfungsi sebagai titik awal program C.
4. **int n1** digunakan untuk menyimpan data variabel **integer** (bilangan desimal).
5. **scanf** digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel yang ditentukan.
6. **printf** yang berfungsi menampilkan output yang telah diketikkan.
7. **%d** : untuk membaca desimal (integer) yang dimasukkan oleh pengguna.
8. **&n1** : berfungsi menyimpan tipe data variabel pada poin **int.**
9. Lalu pada setiap akhir pernyataan ditambahkan **;** untuk menandakan bahwa pernyataan tersebut telah selesai.
10. **if** adalah pernyataan kondisional yang menentukan apakah kode tersebut akan dijalankan berdasarkan kondisi.
11. **baris1[...]** : Menentukan ukuran dari array **satu** dimensi.
12. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
13. **\n** : bermakna newline (baris baru).
14. Terdapat Arithmetic Operator berupa **++, \***.
15. Terdapat juga Comparison Operators berupa : **!=, <.**
16. **return 0** yang berfungsi mengembalikan program kembali seperti semula setelah dieksekusi.
17. Bahasa Python :
18. **n1, n2** berfungsi untuk menyimpan variabel yang disini bersifat desimal (**integer**).
19. **list** : Mengubah objek **map** menjadi **list** yang sebenarnya, sehingga kita bisa bekerja dengan elemen-elemen tersebut seperti dengan **list** pada umumnya.
20. **map** :untuk melakukan operasi pada banyak elemen sekaligus.
21. **int** :digunakan untuk mengonversi tipe data lain menjadi tipe **integer** (bilangan desimal).
22. **input**() : digunakan untuk menerima input dari pengguna, lalu mengembalikan nilai yang dimasukkan sebagai string.
23. **split()** : untuk memisahkan string input yang dihasilkan oleh input() menjadi beberapa bagian berdasarkan spasi.
24. **if** adalah pernyataan kondisional yang menentukan apakah kode tersebut akan dijalankan berdasarkan kondisi.
25. **exit()** : Python akan menganggap program selesai dengan status keluar 0 (menandakan bahwa program selesai tanpa error).
26. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
27. **range** : Menghasilkan urutan angka di dalam tanda ().
28. **list1[...]** : Menentukan ukuran dari array **satu** dimensi.
29. **end=” ”** : untuk mengontrol apa yang dicetak setelah setiap output, disini berfungsi untuk membuat output agar tidak menjalankan newline pada setiap loop.
30. **print** yang berfungsi menampilkan output yang telah diketikkan.
31. Terdapat Arithmetic Operator berupa **\***.
32. Terdapat juga Comparison Operators berupa : **!=.**

# **SOAL 4**

1. Shikamaru merupakan seorang anggota anbu Desa Konoha. Pada saat memata-matai Desa Iwagakure, Shikamaru harus memecahkan sebuah kode untuk memastikan keaslian pesan dari rekannya. Kode merupakan rangkaian karakter yang harus dicocokkan dengan rangkaian karakter yang dimiliki oleh Shikamaru. Karakter yang sama pada kode tersebut akan berubah menjadi tanda bintang (\*) sedangkan karakter yang berbeda akan berubah menjadi tanda (#). Pesan yang asli merupakan pesan yang memiliki jumlah bintang yang lebih dari atau sama dengan jumlah pagar, sedangkan pesan yang palsu merupakan pesan yang memiliki jumlah bintang yang kurang dari jumlah pagar atau panjang karakter tidak sama.

Buatlah sebuah program untuk membantu Shikamaru agar mengetahui pesan yang diterima merupakan pesan asli atau pesan palsu.

Format input:

Baris pertama merupakan kode yang dimiliki oleh Shikamaru

Baris kedua merupakan pesan yang diterima oleh Shikamaru

Format output:

Baris pertama merupakan rangkaian karakter yang telah berubah menjadi tanda bintang (\*) atau pagar (#)

Baris kedua merupakan jumlah tanda bintang

Baris ketiga merupakan jumlah tanda pagar

Baris keempat merupakan pemberitahuan pesan asli atau pesan palsu

|  |  |
| --- | --- |
| **Input** | **Output** |
| Bahasa  Pemrograman | Panjang kalimat berbeda, pesan palsu |
| Ini Pesan Rahasia  1ni p354n Rahas14 | #\*\* ####\* \*\*\*\*\*##  \* = 8 # = 7  Pesan Asli |
| Aku Pasti Bisa  Berjuang lebih | #####\*########  \* = 1  # = 13  Pesan Palsu |

Simpan coding anda dengan nama: **PRAK604-NIM-Nama.py** dan **PRAK604-NIM-Nama.c**

## **Source Code**

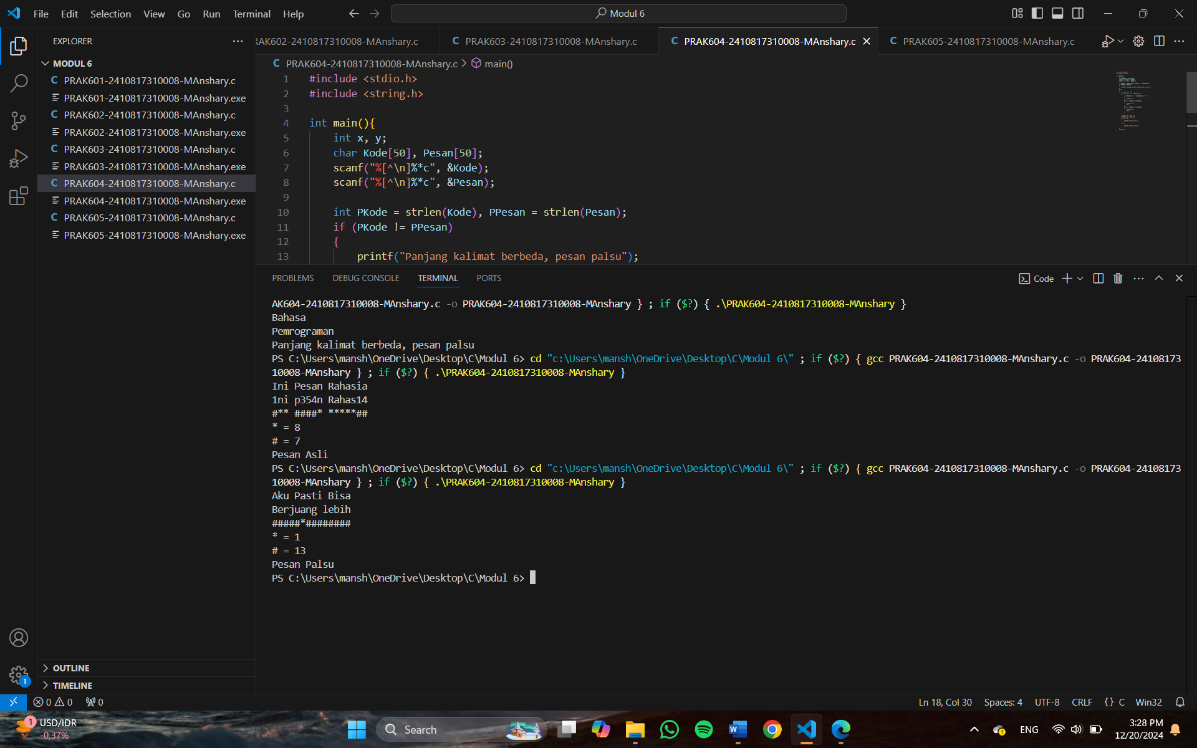
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48 | #include <stdio.h>  #include <string.h>  int main(){      int x, y;      char Kode[50], Pesan[50];      scanf("%[^\n]%\*c", &Kode);      scanf("%[^\n]%\*c", &Pesan);      int PKode = strlen(Kode), PPesan = strlen(Pesan);      if (PKode != PPesan)      {          printf("Panjang kalimat berbeda, pesan palsu");      }      else      {          int x = 0, y = 0;          for (int z = 0; z < PKode; z++)          {              if (Pesan[z] == ' ' && Kode[z] == ' ')              {                  printf(" ");              }              else if (Kode[z] == Pesan[z])              {                  printf("\*");                  x++;              }              else if (Kode[z] != Pesan[z])              {                  printf("#");                  y++;              }          }          printf("\n\* = %d", x);          printf("\n# = %d", y);          if (x >= y)          {              printf("\nPesan Asli");          }          else          {              printf("\nPesan Palsu");          }      }      return 0;  } |

Tabel 1. 7 Source Code Soal 4 Bahasa C

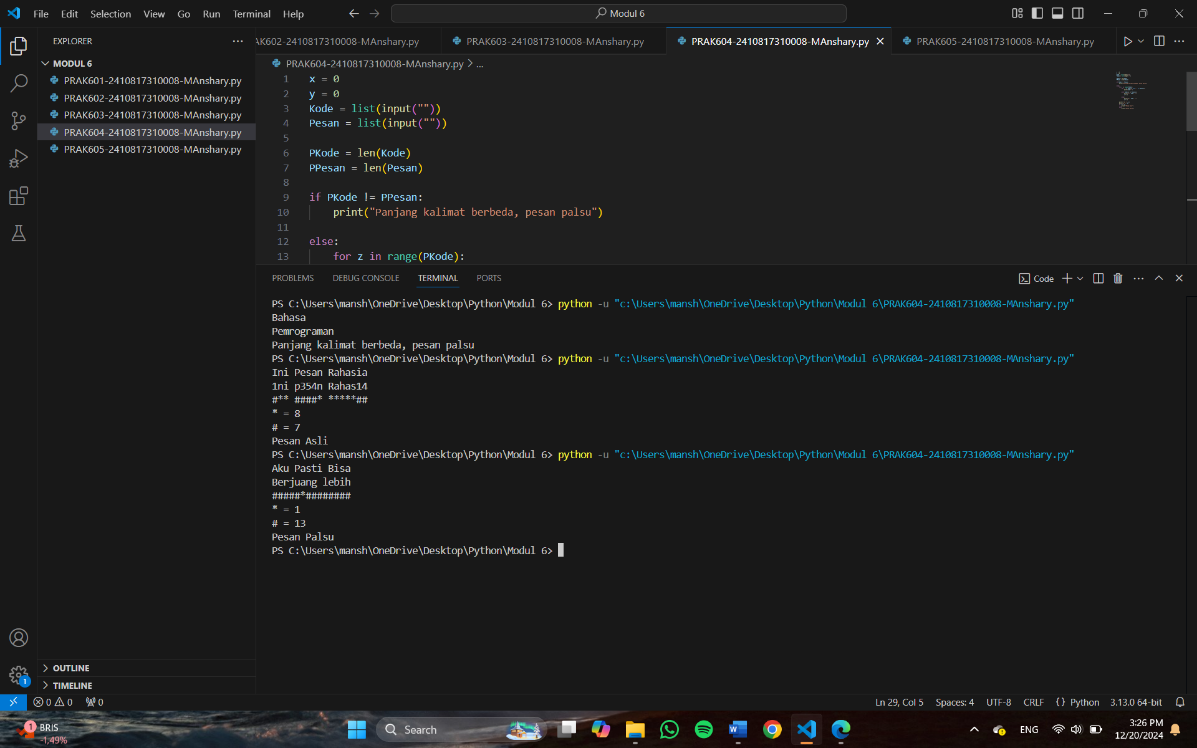
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | x = 0  y = 0  Kode = list(input(""))  Pesan = list(input(""))  PKode = len(Kode)  PPesan = len(Pesan)  if PKode != PPesan:      print("Panjang kalimat berbeda, pesan palsu")  else:      for z in range(PKode):          if " " in Kode[z] and " " in Pesan[z]:              print(end=" ")            elif Kode[z] == Pesan[z]:              print("\*", end = "")              x += 1          else:              print("#", end = "")              y += 1      print("\n\* =", x)      print("# =", y)      if x >= y:          print("Pesan Asli")      else:          print("Pesan Palsu") |

Tabel 1. 8 Source Code Soal 4 Bahasa Python

## **Output Program**



Gambar 1. 7 Screenshot Output Soal 4 Bahasa C



Gambar 1. 8 Screenshot Output Soal 4 Bahasa Python

## **Pembahasan**

1. Bahasa C :
2. **#include<stdio.h>** yang berfungsi untuk menyertakan file header ke dalam program. Salah satu contoh fungsi dari **#include<stdio.h>** yaitu **printf** dan **scanf**.
3. **#include<string.h>** yang berfungsi untuk menyertakan pustaka <**string.h**>, yang berisi prototipe fungsi-fungsi untuk manipulasi string. Salah satu contoh fungsi dari **#include<string.h>** yaitu **strlen.**
4. **int main** yang berfungsi sebagai titik awal program C.
5. **int x** digunakan untuk menyimpan data variabel **integer** (bilangan desimal).
6. **char Kode[50]** berfungsi untuk menyimpan variabel yang bersifat character dengan ukuran Array [50] yang berarti dapat menyimpan hingga 50 karakter.
7. **%[^\n]** : bermakna “baca semua karakter kecuali newline”.
8. **%\*c** :satu karakter (char) akan dibaca tetapi tanda \* menandakan bahwa karakter yang dibaca tidak disimpan.
9. **strlen** : Mengembalikan panjang string (tidak termasuk karakter null).
10. **scanf** digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel yang ditentukan.
11. **printf** yang berfungsi menampilkan output yang telah diketikkan.
12. **%d** : untuk membaca desimal (integer) yang dimasukkan oleh pengguna.
13. **&Kode** : berfungsi menyimpan tipe data variabel pada poin **char.**
14. Lalu pada setiap akhir pernyataan ditambahkan **;** untuk menandakan bahwa pernyataan tersebut telah selesai.
15. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
16. **if, else if, else** adalah pernyataan kondisional yang menentukan apakah kode tersebut akan dijalankan berdasarkan kondisi.
17. **Pesan[...]** : Menentukan ukuran dari array **satu** dimensi.
18. **\n** : bermakna newline (baris baru).
19. Terdapat Logical Operators && (AND).
20. Terdapat Arithmetic Operator berupa **++.**
21. Terdapat juga Comparison Operators berupa : !=, <, **= =, >=**.
22. **return 0** yang berfungsi mengembalikan program kembali seperti semula setelah dieksekusi.
23. Bahasa Python :
24. **Kode** berfungsi untuk menyimpan variabel yang disini bersifat string lalu mengonversi string tersebut menjadi list.
25. **len** : Menghitung jumlah elemen dalam list elemen.
26. **list** : Mengubah objek **map** menjadi **list** yang sebenarnya, sehingga kita bisa bekerja dengan elemen-elemen tersebut seperti dengan **list** pada umumnya.
27. **map** :untuk melakukan operasi pada banyak elemen sekaligus.
28. **int** :digunakan untuk mengonversi tipe data lain menjadi tipe **integer** (bilangan desimal).
29. **input**() : digunakan untuk menerima input dari pengguna, lalu mengembalikan nilai yang dimasukkan sebagai string.
30. **split()** : untuk memisahkan string input yang dihasilkan oleh input() menjadi beberapa bagian berdasarkan spasi.
31. **if, elif, else** adalah pernyataan kondisional yang menentukan apakah kode tersebut akan dijalankan berdasarkan kondisi.
32. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
33. **range** : Menghasilkan urutan angka di dalam tanda ().
34. **Kode[...]** : Menentukan ukuran dari array **satu** dimensi.
35. **end=” ”** : untuk mengontrol apa yang dicetak setelah setiap output, disini berfungsi untuk membuat output agar tidak menjalankan newline pada setiap loop.
36. **\n** : bermakna newline (baris baru).
37. **print** yang berfungsi menampilkan output yang telah diketikkan.
38. Terdapat Arithmetic Operator berupa **+, \*, %**.
39. Terdapat Assignment Operators berupa **+=**.
40. Terdapat Logical operators berupa **and**.
41. Terdapat juga Comparison Operators berupa !=, = =, >=**.**

# **SOAL 5**

1. Buatlah program untuk menghitung hasil dari perkalian 2 buah matriks persegi.

Format Input :

Baris pertama merupakan ordo matriks n×n

Baris kedua merupakan isi matriks A

Baris ketiga merupakan isi matriks B

Format output:

Output merupakan hasil kali dari matriks A×B

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  Matriks A  1 2  3 4  Matriks B  1 2  3 4 | Matriks AXB  7 10  15 22 |
| 3  Matriks A  1 2 3  3 4 5  6 7 8  Matriks B  1 2 3  3 4 5  6 7 8 | Matriks AXB  25 31 37  45 57 69  75 96 117 |

Simpan coding anda dengan nama: **PRAK605-NIM-Nama.py** dan **PRAK605-NIM-Nama.c**

## **Source Code**

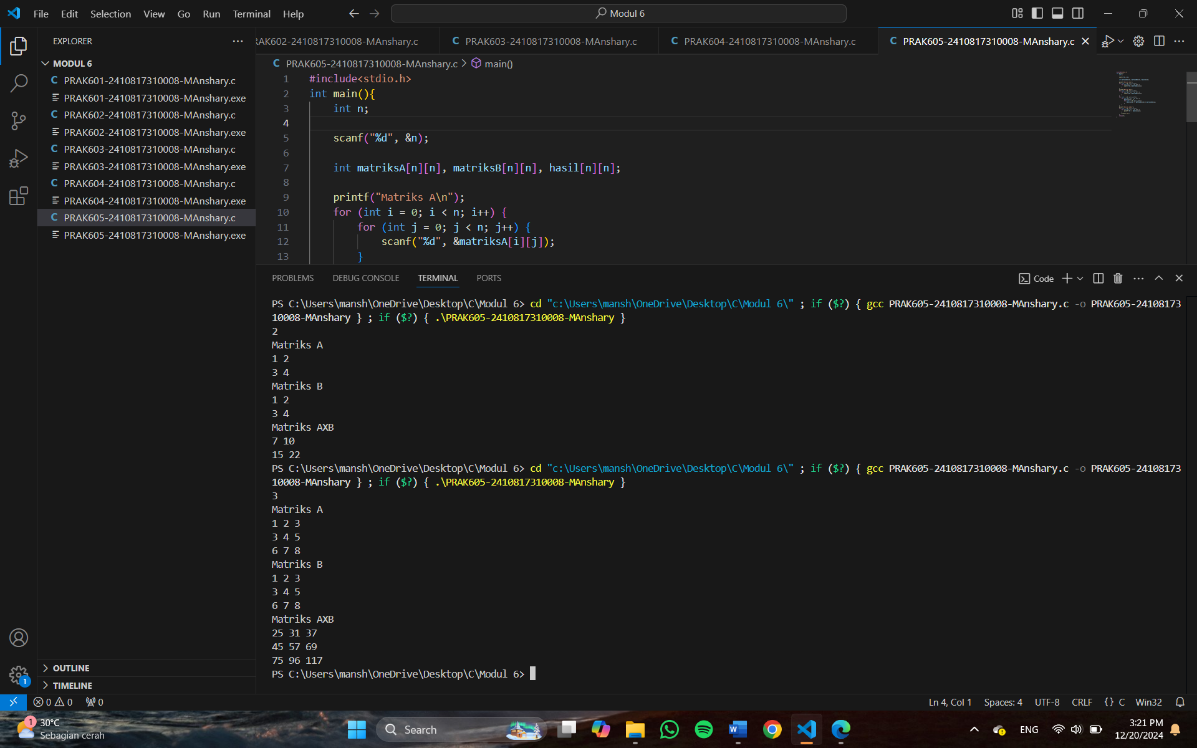
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37 | #include<stdio.h>  int main(){      int n;      scanf("%d", &n);      int matriksA[n][n], matriksB[n][n], hasil[n][n];      printf("Matriks A\n");      for (int i = 0; i < n; i++) {          for (int j = 0; j < n; j++) {              scanf("%d", &matriksA[i][j]);          }      }      printf("Matriks B\n");      for (int i = 0; i < n; i++) {          for (int j = 0; j < n; j++) {              scanf("%d", &matriksB[i][j]);          }      }      for (int i = 0; i < n; i++) {          for (int j = 0; j < n; j++) {              hasil[i][j] = 0;              for (int k = 0; k < n; k++) {                  hasil[i][j] += matriksA[i][k] \* matriksB[k][j];              }          }      }      printf("Matriks AXB\n");      for (int i = 0; i < n; i++) {          for (int j = 0; j < n; j++) {              printf("%d ", hasil[i][j]);          }          printf("\n");      }      return 0;  } |

Tabel 1. 9 Source Code Soal 5 Bahasa C

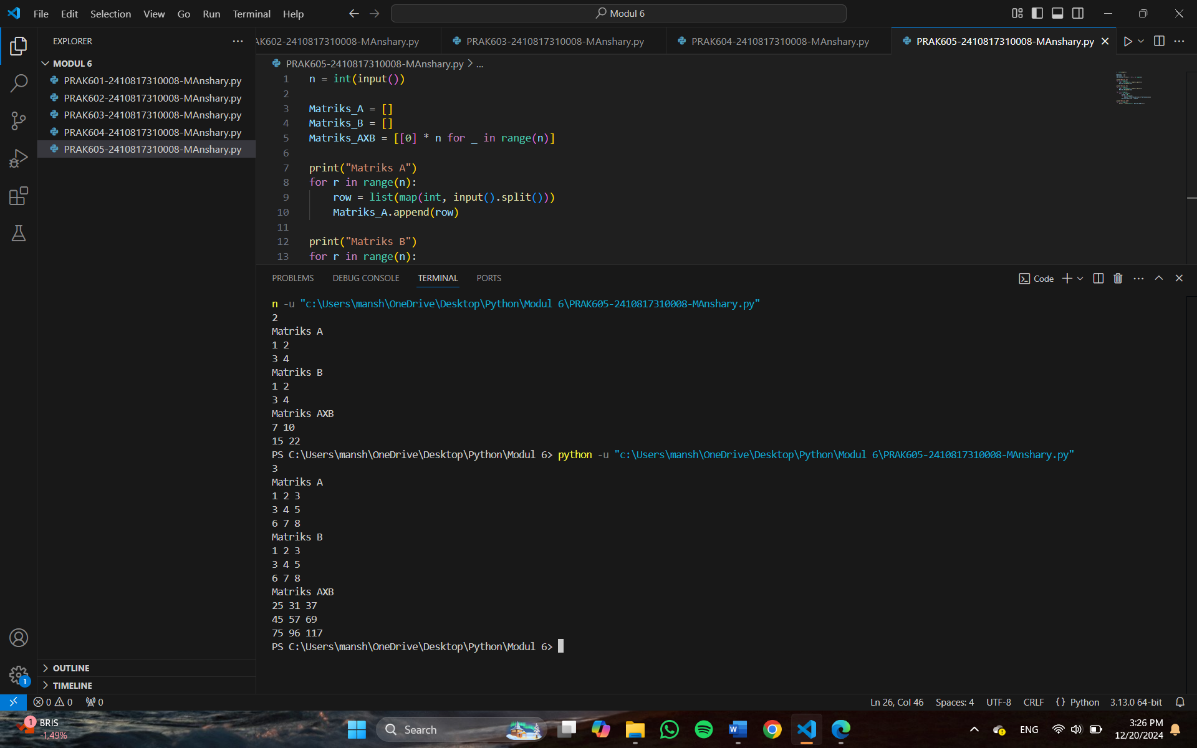
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | n = int(input())  Matriks\_A = []  Matriks\_B = []  Matriks\_AXB = [[0] \* n for \_ in range(n)]  print("Matriks A")  for r in range(n):      row = list(map(int, input().split()))      Matriks\_A.append(row)  print("Matriks B")  for r in range(n):      row = list(map(int, input().split()))      Matriks\_B.append(row)  for r in range(n):      for y in range(n):          jumlah = 0          for a in range(n):              jumlah += Matriks\_A[r][a] \* Matriks\_B[a][y]          Matriks\_AXB[r][y] = jumlah  print("Matriks AXB")  for r in range(n):      print(" ".join(map(str, Matriks\_AXB[r]))) |

Tabel 1. 10 Source Code Soal 5 Bahasa Python

## **Output Program**



Gambar 1. 9 Screenshot Output Soal 5 Bahasa C



Gambar 1. 10 Screenshot Output Soal 5 Bahasa Python

## **Pembahasan**

1. Bahasa C :
2. **#include<stdio.h>** yang berfungsi untuk menyertakan file header ke dalam program. Salah satu contoh fungsi dari **#include<stdio.h>** yaitu **printf** dan **scanf**.
3. **int main** yang berfungsi sebagai titik awal program C.
4. **int n** digunakan untuk menyimpan data variabel **integer** (bilangan desimal).
5. **scanf** digunakan untuk membaca input dari pengguna dan menyimpannya dalam variabel yang ditentukan.
6. **printf** yang berfungsi menampilkan output yang telah diketikkan.
7. **matriksA[...][...]** : Menentukan ukuran dari array **dua** dimensi.
8. **%d** : untuk membaca desimal (integer) yang dimasukkan oleh pengguna.
9. **&n** : berfungsi menyimpan tipe data variabel pada poin **int.**
10. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
11. Lalu pada setiap akhir pernyataan ditambahkan **;** untuk menandakan bahwa pernyataan tersebut telah selesai.
12. **\n** : bermakna newline (baris baru).
13. Terdapat Arithmetic Operator berupa ++, \*.
14. Terdapat Assignment Operators berupa **+=**.
15. **return 0** yang berfungsi mengembalikan program kembali seperti semula setelah dieksekusi.
16. Bahasa Python :
17. **n** berfungsi untuk menyimpan variabel yang disini bersifat desimal (**integer**).
18. **list** : Mengubah objek **map** menjadi **list** yang sebenarnya, sehingga kita bisa bekerja dengan elemen-elemen tersebut seperti dengan **list** pada umumnya.
19. **map** :untuk melakukan operasi pada banyak elemen sekaligus.
20. **int** :digunakan untuk mengonversi tipe data lain menjadi tipe **integer** (bilangan desimal).
21. **input**() : digunakan untuk menerima input dari pengguna, lalu mengembalikan nilai yang dimasukkan sebagai string.
22. **for** : sebuah struktur pengulangan (loop) yang digunakan untuk menjalankan sekelompok perintah secara berulang berdasarkan kondisi tertentu.
23. **range** : Menghasilkan urutan angka di dalam tanda ().
24. **append()** : metode yang digunakan untuk menambahkan elemen ke dalam **list** yang sudah ada.
25. **Matriks\_A[...][...]** : Menentukan ukuran dari array **dua** dimensi.
26. **join()** : digunakan untuk menggabungkan elemen-elemen dalam iterable (seperti list, tuple, atau string) menjadi sebuah string tunggal, dengan setiap elemen dipisahkan oleh string yang ada sebelum **join(),** Dalam hal ini, **" "** adalah string pemisah yang digunakan untuk menggabungkan elemen-elemen tersebut dengan **spasi**.
27. **str** : fungsi yang digunakan untuk mengonversi objek menjadi string.
28. **print** yang berfungsi menampilkan output yang telah diketikkan.
29. Terdapat Arithmetic Operator berupa\*.
30. Terdapat Assignment Operators berupa **+=**.