**LAPORAN AKHIR PRAKTIKUM**

**PEMROGRAMAN I**

****

**Oleh:**

**Aulia Az Zahra NIM. 2410817120021**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**DESEMBER 2024**

# LEMBAR PENGESAHAN

**LAPORAN AKHIR PRAKTIKUM PEMROGRAMAN I**

Laporan Praktikum Pemrograman I

Modul 0 : How To Program

Modul 1 : Variable, Tipe Data dan Operator

Modul 2 : Input dan Output

Modul 3 : Kondisional

Modul 4 : Loop

Modul 5 : Fungsi

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

Nama Praktikan : Aulia Az Zahra

NIM : 2410817120021

|  |  |
| --- | --- |
| 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](#_Toc186490250)

[DAFTAR ISI 3](#_Toc186490251)

[DAFTAR TABEL 9](#_Toc186490252)

[DAFTAR GAMBAR 18](#_Toc186490253)

[MODUL 0: HOW TO PROGRAM 21](#_Toc186490254)

[SOAL 1 21](#_Toc186490255)

[A. Source Code 21](#_Toc186490256)

[B. Output Program 22](#_Toc186490257)

[Pembahasan 24](#_Toc186490258)

[SOAL 2 25](#_Toc186490259)

[A. Source Code 25](#_Toc186490260)

[B. Output Program 26](#_Toc186490261)

[C. Pembahasan 27](#_Toc186490262)

[SOAL 3 28](#_Toc186490263)

[A. Source Code 28](#_Toc186490264)

[B. Output Program 29](#_Toc186490265)

[C. Pembahasan 30](#_Toc186490266)

[SOAL 4 31](#_Toc186490267)

[A. Source Code 32](#_Toc186490268)

[B. Output Program 33](#_Toc186490269)

[C. Pembahasan 34](#_Toc186490270)

[SOAL 5 35](#_Toc186490271)

[A. Source Code 36](#_Toc186490272)

[B. Output Program 37](#_Toc186490273)

[C. Pembahasan 38](#_Toc186490274)

[MODUL 1 : Variabel, Tipe Data dan Operator 39](#_Toc186490275)

[SOAL 1 39](#_Toc186490276)

[A. Source Code 40](#_Toc186490277)

[B. Output Program 41](#_Toc186490278)

[C. Pembahasan 43](#_Toc186490279)

[SOAL 2 44](#_Toc186490280)

[A. Source Code 45](#_Toc186490281)

[B. Output Program 46](#_Toc186490282)

[C. Pembahasan 47](#_Toc186490283)

[SOAL 3 49](#_Toc186490284)

[A. Source Code 50](#_Toc186490285)

[B. Output Program 51](#_Toc186490286)

[C. Pembahasan 52](#_Toc186490287)

[SOAL 4 54](#_Toc186490288)

[A. Source Code 55](#_Toc186490289)

[B. Output Program 56](#_Toc186490290)

[C. Pembahasan 57](#_Toc186490291)

[SOAL 5 59](#_Toc186490292)

[A. Source Code 60](#_Toc186490293)

[B. Output Program 62](#_Toc186490294)

[C. Pembahasan 63](#_Toc186490295)

[SOAL 6 65](#_Toc186490296)

[A. Source Code 66](#_Toc186490297)

[B. Output Program 68](#_Toc186490298)

[C. Pembahasan 69](#_Toc186490299)

[SOAL 7 70](#_Toc186490300)

[A. Source Code 71](#_Toc186490301)

[B. Output Program 73](#_Toc186490302)

[C. Pembahasan 74](#_Toc186490303)

[SOAL 8 76](#_Toc186490304)

[A. Source Code 77](#_Toc186490305)

[B. Output Program 79](#_Toc186490306)

[C. Pembahasan 80](#_Toc186490307)

[SOAL 9 82](#_Toc186490308)

[A. Source Code 83](#_Toc186490309)

[B. Output Program 84](#_Toc186490310)

[C. Pembahasan 85](#_Toc186490311)

[SOAL 10 87](#_Toc186490312)

[A. Source Code 88](#_Toc186490313)

[B. Output Program 90](#_Toc186490314)

[C. Pembahasan 91](#_Toc186490315)

[MODUL 2 : Input dan Output 93](#_Toc186490316)

[SOAL 1 93](#_Toc186490317)

[A. Source Code 94](#_Toc186490318)

[B. Output Program 96](#_Toc186490319)

[C. Pembahasan 97](#_Toc186490320)

[SOAL 2 99](#_Toc186490321)

[A. Source Code 100](#_Toc186490322)

[B. Output Program 101](#_Toc186490323)

[C. Pembahasan 102](#_Toc186490324)

[SOAL 3 104](#_Toc186490325)

[A. Source Code 105](#_Toc186490326)

[A. Output Program 106](#_Toc186490327)

[B. Pembahasan 107](#_Toc186490328)

[SOAL 4 109](#_Toc186490329)

[A. Source Code 110](#_Toc186490330)

[B. Output Program 112](#_Toc186490331)

[C. Pembahasan 113](#_Toc186490332)

[SOAL 5 115](#_Toc186490333)

[A. Source Code 116](#_Toc186490334)

[B. Output Program 118](#_Toc186490335)

[C. Pembahasan 119](#_Toc186490336)

[MODUL 3 : Kondisional 121](#_Toc186490337)

[SOAL 1 121](#_Toc186490338)

[A. Source Code 122](#_Toc186490339)

[B. Output Program 124](#_Toc186490340)

[C. Pembahasan 125](#_Toc186490341)

[SOAL 2 127](#_Toc186490342)

[A. Source Code 128](#_Toc186490343)

[B. Output Program 130](#_Toc186490344)

[C. Pembahasan 131](#_Toc186490345)

[SOAL 3 133](#_Toc186490346)

[A. Source Code 134](#_Toc186490347)

[B. Output Program 136](#_Toc186490348)

[C. Pembahasan 137](#_Toc186490349)

[SOAL 4 139](#_Toc186490350)

[A. Source Code 140](#_Toc186490351)

[B. Output Program 142](#_Toc186490352)

[C. Pembahasan 143](#_Toc186490353)

[SOAL 5 145](#_Toc186490354)

[A. Source Code 146](#_Toc186490355)

[B. Output Program 149](#_Toc186490356)

[C. Pembahasan 150](#_Toc186490357)

[MODUL 4 : Loop 152](#_Toc186490358)

[SOAL 1 152](#_Toc186490359)

[A. Source Code 154](#_Toc186490360)

[B. Output Program 156](#_Toc186490361)

[C. Pembahasan 157](#_Toc186490362)

[SOAL 2 159](#_Toc186490363)

[A. Source Code 160](#_Toc186490364)

[B. Output Program 162](#_Toc186490365)

[C. Pembahasan 163](#_Toc186490366)

[SOAL 3 165](#_Toc186490367)

[A. Source Code 166](#_Toc186490368)

[B. Output Program 168](#_Toc186490369)

[C. Pembahasan 169](#_Toc186490370)

[SOAL 4 171](#_Toc186490371)

[A. Source Code 173](#_Toc186490372)

[B. Output Program 176](#_Toc186490373)

[C. Pembahasan 177](#_Toc186490374)

[SOAL 5 179](#_Toc186490375)

[A. Source Code 180](#_Toc186490376)

[B. Output Program 182](#_Toc186490377)

[C. Pembahasan 183](#_Toc186490378)

[MODUL 5 : Fungsi 185](#_Toc186490379)

[SOAL 1 185](#_Toc186490380)

[A. Source Code 186](#_Toc186490381)

[B. Output Program 188](#_Toc186490382)

[C. Pembahasan 189](#_Toc186490383)

[SOAL 2 191](#_Toc186490384)

[A. Source Code 193](#_Toc186490385)

[B. Output Program 195](#_Toc186490386)

[C. Pembahasan 196](#_Toc186490387)

[SOAL 3 198](#_Toc186490388)

[A. Source Code 200](#_Toc186490389)

[B. Output Program 203](#_Toc186490390)

[C. Pembahasan 204](#_Toc186490391)

[SOAL 4 206](#_Toc186490392)

[A. Source Code 208](#_Toc186490393)

[B. Output Program 210](#_Toc186490394)

[C. Pembahasan 211](#_Toc186490395)

[SOAL 5 213](#_Toc186490396)

[A. Source Code 215](#_Toc186490397)

[B. Output Program 217](#_Toc186490398)

[C. Pembahasan 218](#_Toc186490399)

[MODUL 6 : Array 220](#_Toc186490400)

[SOAL 1 220](#_Toc186490401)

[A. Source Code 222](#_Toc186490402)

[B. Output Program 224](#_Toc186490403)

[C. Pembahasan 225](#_Toc186490404)

[SOAL 2 227](#_Toc186490405)

[A. Source Code 228](#_Toc186490406)

[B. Output Program 230](#_Toc186490407)

[C. Pembahasan 231](#_Toc186490408)

[SOAL 3 233](#_Toc186490409)

[A. Source Code 234](#_Toc186490410)

[B. Output Program 236](#_Toc186490411)

[C. Pembahasan 237](#_Toc186490412)

[SOAL 4 239](#_Toc186490413)

[A. Source Code 241](#_Toc186490414)

[B. Output Program 245](#_Toc186490415)

[C. Pembahasan 246](#_Toc186490416)

[SOAL 5 248](#_Toc186490417)

[A. Source Code 250](#_Toc186490418)

[B. Output Program 253](#_Toc186490419)

[C. Pembahasan 254](#_Toc186490420)

[RANGKUMAN 256](#_Toc186490421)

[TAUTAN GIT 259](#_Toc186490422)

# DAFTAR TABEL

**Modul 0 : How To Program**

[Table 1 Source Code Bahasa C Soal 1 15](#_Toc186490723)

[Table 2 Source Code Bahasa Python Soal 1 15](#_Toc186490724)

[Table 3 Source Code Bahasa C Soal 2 19](#_Toc186490725)

[Table 4 Source Code Bahasa Python Soal 2 19](#_Toc186490726)

[Table 5 Source Code Bahasa C Soal 3 22](#_Toc186490727)

[Table 6 Source Code Bahasa Python Soal 3 22](#_Toc186490728)

[Table 7 Source Code Bahasa C Soal 4 26](#_Toc186490729)

[Table 8 Source Code Bahasa Python Soal 4 26](#_Toc186490730)

[Table 9 Source Code Bahasa C Soal 5 30](#_Toc186490731)

[Table 10 Source Code Bahasa Python Soal 5 30](#_Toc186490732)

**Modul 1 : Variabel, Tipe Data dan Operator**

[Table 11 Source Code Bahasa C Soal 1 34](#_Toc186490733)

[Table 12 Source Code Bahasa Python Soal 1 34](#_Toc186490734)

[Table 13 Source Code Bahasa C Soal 2 39](#_Toc186490735)

[Table 14 Source Code Bahasa Python Soal 2 39](#_Toc186490736)

[Table 15 Source Code Bahasa C Soal 3 44](#_Toc186490737)

[Table 16 Source Code Bahasa Python Soal 3 44](#_Toc186490738)

[Table 17 Source Code Bahasa C Soal 4 49](#_Toc186490739)

[Table 18 Source Code Bahasa Python Soal 4 49](#_Toc186490740)

[Table 19 Source Code Bahasa C Soal 5 54](#_Toc186490741)

[Table 20 Source Code Bahasa Python Soal 5 55](#_Toc186490742)

[Table 21 Source Code Bahasa C Soal 6 60](#_Toc186490743)

[Table 22 Source Code Bahasa Python Soal 6 61](#_Toc186490744)

[Table 23 Source Code Bahasa C Soal 7 65](#_Toc186490745)

[Table 24 Source Code Bahasa Python Soal 66](#_Toc186490746)

[Table 25 Source Code Bahasa C Soal 8 71](#_Toc186490747)

[Table 26 Source Code Bahasa Python Soal 8 72](#_Toc186490748)

[Table 27 Source Code Bahasa C Soal 9 77](#_Toc186490749)

[Table 28 Source Code Bahasa Python Soal 9 77](#_Toc186490750)

[Table 29 Source Code Bahasa C Soal 10 82](#_Toc186490751)

[Table 30 Source Code Bahasa Python Soal 10 83](#_Toc186490752)

**Modul 2 : Input dan Output**

[Table 31 Source Code Bahasa C Soal 1 89](#_Toc186490753)

[Table 32 Source Code Bahasa Python Soal 1 89](#_Toc186490754)

[Table 33 Source Code Bahasa C Soal 2 94](#_Toc186490755)

[Table 34 Source Code Bahasa Python Soal 2 94](#_Toc186490756)

[Table 35 Source Code Bahasa C Soal 3 99](#_Toc186490757)

[Table 36 Source Code Bahasa Python Soal 3 99](#_Toc186490758)

[Table 37 Source Code Bahasa C Soal 4 104](#_Toc186490759)

[Table 38 Source Code Bahasa Python Soal 4 105](#_Toc186490760)

[Table 39 Source Code Bahasa C Soal 5 110](#_Toc186490761)

[Table 40 Source Code Bahasa Python Soal 5 111](#_Toc186490762)

**Modul 3 : Kondisional**

[Table 41 Source Code Bahasa C Soal 1 116](#_Toc186490763)

[Table 42 Source Code Bahasa Python Soal 1 117](#_Toc186490764)

[Table 43 Source Code Bahasa C Soal 2 123](#_Toc186490765)

[Table 44 Source Code Bahasa Python Soal 2 123](#_Toc186490766)

[Table 45 Source Code Bahasa C Soal 3 128](#_Toc186490767)

[Table 46 Source Code Bahasa Python Soal 3 129](#_Toc186490768)

[Table 47 Source Code Bahasa C Soal 4 135](#_Toc186490769)

[Table 48 Source Code Bahasa Python Soal 4 135](#_Toc186490770)

[Table 49 Source Code Bahasa C Soal 5 141](#_Toc186490771)

[Table 50 Source Code Bahasa Python Soal 5 142](#_Toc186490772)

**Modul 4 : Loop**

[Table 51 Source Code Bahasa C Soal 1 148](#_Toc186490773)

[Table 52 Source Code Bahasa Python Soal 1 149](#_Toc186490774)

[Table 53 Source Code Bahasa C Soal 2 154](#_Toc186490775)

[Table 54 Source Code Bahasa Python Soal 2 155](#_Toc186490776)

[Table 55 Source Code Bahasa C Soal 3 161](#_Toc186490777)

[Table 56 Source Code Bahasa Python Soal 3 161](#_Toc186490778)

[Table 57 Source Code Bahasa C Soal 4 168](#_Toc186490779)

[Table 58 Source Code Bahasa Python Soal 4 169](#_Toc186490780)

[Table 59 Source Code Bahasa C Soal 5 174](#_Toc186490781)

[Table 60 Source Code Bahasa Python Soal 5 175](#_Toc186490782)

**Modul 5 : Fungsi**

[Table 61 Source Code Bahasa C Soal 1 181](#_Toc186490783)

[Table 62 Source Code Bahasa Python Soal 1 181](#_Toc186490784)

[Table 63 Source Code Bahasa C Soal 2 187](#_Toc186490785)

[Table 64 Source Code Bahasa Python Soal 2 188](#_Toc186490786)

[Table 65 Source Code Bahasa C Soal 3 195](#_Toc186490787)

[Table 66 Source Code Bahasa Python Soal 3 196](#_Toc186490788)

[Table 67 Source Code Bahasa C Soal 4 202](#_Toc186490789)

[Table 68 Source Code Bahasa Python Soal 4 203](#_Toc186490790)

[Table 69 Source Code Bahasa C Soal 5 209](#_Toc186490791)

[Table 70 Source Code Bahasa Python Soal 5 210](#_Toc186490792)

**Modul 6 : Array**

[Table 71 Source Code Bahasa C Soal 1 216](#_Toc186490793)

[Table 72 Source Code Bahasa Python Soal 1 217](#_Toc186490794)

[Table 73 Source Code Bahasa C Soal 2 222](#_Toc186490795)

[Table 74 Source Code Bahasa Python Soal 2 223](#_Toc186490796)

[Table 75 Source Code Bahasa C Soal 3 228](#_Toc186490797)

[Table 76 Source Code Bahasa Python Soal 3 229](#_Toc186490798)

[Table 77 Source Code Bahasa C Soal 4 236](#_Toc186490799)

[Table 78 Source Code Bahasa Python Soal 4 238](#_Toc186490800)

[Table 79 Source Code Bahasa C Soal 5 245](#_Toc186490801)

[Table 80 Source Code Bahasa Python Soal 5 246](#_Toc186490802)

# DAFTAR GAMBAR

**MODUL 0: HOW TO PROGRAM**

[Gambar 1 Screenshot Output Bahasa C Soal 1 17](#_Toc186477343)

[Gambar 2 Screenshot Output Bahasa Python Soal 1 18](#_Toc186477344)

[Gambar 4 Screenshot Output Bahasa Python Soal 2 21](#_Toc186477345)

[Gambar 3 Screenshot Output Bahasa C Soal 2 21](#_Toc186477346)

[Gambar 5 Screenshot Output Bahasa C Soal 3 24](#_Toc186477347)

[Gambar 6 Screenshot Output Bahasa Python Soal 3 24](#_Toc186477348)

[Gambar 7 Screenshot Output Bahasa C Soal 4 28](#_Toc186477349)

[Gambar 8 Screenshot Output Bahasa Python Soal 4 28](#_Toc186477350)

[Gambar 9 Screenshot Output Bahasa C Soal 5 32](#_Toc186477351)

[Gambar 10 Screenshot Output Bahasa Python Soal 5 32](#_Toc186477352)

**MODUL 1 : Variabel, Tipe Data dan Operator**

[Gambar 11 Screenshot Output Bahasa C Soal 1 36](#_Toc186477353)

[Gambar 12 Screenshot Output Bahasa Python Soal 1 37](#_Toc186477354)

[Gambar 13 Screenshot Output Bahasa C Soal 2 41](#_Toc186477355)

[Gambar 14 Screenshot Output Bahasa Python Soal 2 41](#_Toc186477356)

[Gambar 15 Screenshot Output Bahasa C Soal 3 46](#_Toc186477357)

[Gambar 16 Screenshot Output Bahasa Python Soal 3 46](#_Toc186477358)

[Gambar 17 Screenshot Output Bahasa C Soal 4 51](#_Toc186477359)

[Gambar 18 Screenshot Output Bahasa Python Soal 4 51](#_Toc186477360)

[Gambar 19 Screenshot Output Bahasa C Soal 5 57](#_Toc186477361)

[Gambar 20 Screenshot Output Bahasa Python Soal 5 57](#_Toc186477362)

[Gambar 21 Screenshot Output Bahasa C Soal 6 63](#_Toc186477363)

[Gambar 22 Screenshot Output Bahasa Python Soal 6 63](#_Toc186477364)

[Gambar 23 Screenshot Output Bahasa C Soal 7 68](#_Toc186477365)

[Gambar 24 Screenshot Output Bahasa Python Soal 7 68](#_Toc186477366)

[Gambar 25 Screenshot Output Bahasa C Soal 8 74](#_Toc186477367)

[Gambar 26 Screenshot Output Bahasa Python Soal 8 74](#_Toc186477368)

[Gambar 27 Screenshot Output Bahasa C Soal 9 79](#_Toc186477369)

[Gambar 28 Screenshot Output Bahasa Python Soal 9 79](#_Toc186477370)

[Gambar 29 Screenshot Output Bahasa C Soal 10 85](#_Toc186477371)

[Gambar 30 Screenshot Output Bahasa Python Soal 10 86](#_Toc186477372)

**MODUL 2 : Input dan Output**

[Gambar 31 Screenshot Output Bahasa C Soal 1 91](#_Toc186477373)

[Gambar 32 Screenshot Output Bahasa Python Soal 1 91](#_Toc186477374)

[Gambar 33 Screenshot Output Bahasa C Soal 2 96](#_Toc186477375)

[Gambar 34 Screenshot Output Bahasa Python Soal 2 96](#_Toc186477376)

[Gambar 35 Screenshot Output Bahasa C Soal 3 101](#_Toc186477377)

[Gambar 36 Screenshot Output Bahasa Python Soal 3 101](#_Toc186477378)

[Gambar 37 Screenshot Output Bahasa C Soal 4 107](#_Toc186477379)

[Gambar 38 Screenshot Output Bahasa Python Soal 4 107](#_Toc186477380)

[Gambar 39 Screenshot Output Bahasa C Soal 5 113](#_Toc186477381)

[Gambar 40 Screenshot Output Bahasa Python Soal 5 113](#_Toc186477382)

**MODUL 3 : Kondisional**

[Gambar 41 Screenshot Output Bahasa C Soal 1 119](#_Toc186477383)

[Gambar 42 Screenshot Output Bahasa Python Soal 1 119](#_Toc186477384)

[Gambar 43 Screenshot Output Bahasa C Soal 2 125](#_Toc186477385)

[Gambar 44 Screenshot Output Bahasa Python Soal 2 125](#_Toc186477386)

[Gambar 45 Screenshot Output Bahasa C Soal 3 131](#_Toc186477387)

[Gambar 46 Screenshot Output Bahasa Python Soal 3 131](#_Toc186477388)

[Gambar 47 Screenshot Output Bahasa C Soal 4 137](#_Toc186477389)

[Gambar 48 Screenshot Output Bahasa Python Soal 4 137](#_Toc186477390)

[Gambar 49 Screenshot Output Bahasa C Soal 5 144](#_Toc186477391)

[Gambar 50 Screenshot Output Bahasa Python Soal 5 144](#_Toc186477392)

**MODUL 4 : Loop**

[Gambar 51 Screenshot Output Bahasa C Soal 1 151](#_Toc186477393)

[Gambar 52 Screenshot Output Bahasa Python Soal 1 151](#_Toc186477394)

[Gambar 53 Screenshot Output Bahasa C Soal 2 157](#_Toc186477395)

[Gambar 54 Screenshot Output Bahasa Python Soal 2 157](#_Toc186477396)

[Gambar 55 Screenshot Output Bahasa C Soal 3 163](#_Toc186477397)

[Gambar 56 Screenshot Output Bahasa Python Soal 3 163](#_Toc186477398)

[Gambar 57 Screenshot Output Bahasa C Soal 4 171](#_Toc186477399)

[Gambar 58 Screenshot Output Bahasa Python Soal 4 171](#_Toc186477400)

[Gambar 59 Screenshot Output Bahasa C Soal 5 177](#_Toc186477401)

[Gambar 60 Screenshot Output Bahasa Python Soal 5 177](#_Toc186477402)

**MODUL 5 : Fungsi**

[Gambar 61 Screenshot Output Bahasa C Soal 1 183](#_Toc186477403)

[Gambar 62 Screenshot Output Bahasa Python Soal 1 183](#_Toc186477404)

[Gambar 63 Screenshot Output Bahasa C Soal 2 190](#_Toc186477405)

[Gambar 64 Screenshot Output Bahasa Python Soal 2 190](#_Toc186477406)

[Gambar 65 Screenshot Output Bahasa C Soal 3 198](#_Toc186477407)

[Gambar 66 Screenshot Output Bahasa Python Soal 3 198](#_Toc186477408)

[Gambar 67 Screenshot Output Bahasa C Soal 4 205](#_Toc186477409)

[Gambar 68 Screenshot Output Bahasa Python Soal 4 205](#_Toc186477410)

[Gambar 69 Screenshot Output Bahasa C Soal 5 212](#_Toc186477411)

[Gambar 70 Screenshot Output Bahasa Python Soal 5 212](#_Toc186477412)

**MODUL 6 : Array**

[Gambar 71 Screenshot Output Bahasa C Soal 1 219](#_Toc186477413)

[Gambar 72 Screenshot Output Bahasa Python Soal 1 219](#_Toc186477414)

[Gambar 73 Screenshot Output Bahasa C Soal 2 225](#_Toc186477415)

[Gambar 74 Screenshot Output Bahasa Python Soal 2 225](#_Toc186477416)

[Gambar 75 Screenshot Output Bahasa C Soal 3 231](#_Toc186477417)

[Gambar 76 Screenshot Output Bahasa Python Soal 3 231](#_Toc186477418)

[Gambar 77 Screenshot Output Bahasa C Soal 4 240](#_Toc186477419)

[Gambar 78 Screenshot Output Bahasa Python Soal 4 240](#_Toc186477420)

[Gambar 79 Screenshot Output Bahasa C Soal 5 248](#_Toc186477421)

[Gambar 80 Screenshot Output Bahasa Python Soal 5 248](#_Toc186477422)

# MODUL 0: HOW TO PROGRAM

# SOAL 1

1. Buatlah program yang dapat menghasilkan output sebagaimana berikut:

|  |
| --- |
| **Output** |
| Saya Calon Programmer No. 1 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5 | #include <stdio.h>  int main()  {     printf("Saya Calon Programmer No. 1");  } |

Table 1 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1 | print ("Saya Calon Programmer No. 1") |

Table 2 Source Code Bahasa Python Soal 1

## Sebuah gambar berisi teks, cuplikan layar, software, Software multimedia Deskripsi dibuat secara otomatisOutput Program

Gambar 1 Screenshot Output Bahasa C Soal 1



Gambar 2 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C
   * Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
   * Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().*
   * Pada baris 4: syntax *printf (“ “);* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan.
2. Pembahasan Bahasa Python

* Pada baris 1: syntax print() berfungsi untuk menampilkan output ke layer. Syntax (””) digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 2

1. Buatlah program yang dapat menghasilkan output sebagaimana berikut:

|  |
| --- |
| **Output** |
| Selamat Pagi, Nama Anda  Selamat Siang, Nama Anda  Selamat Malam, Nama Anda |

## Source Code

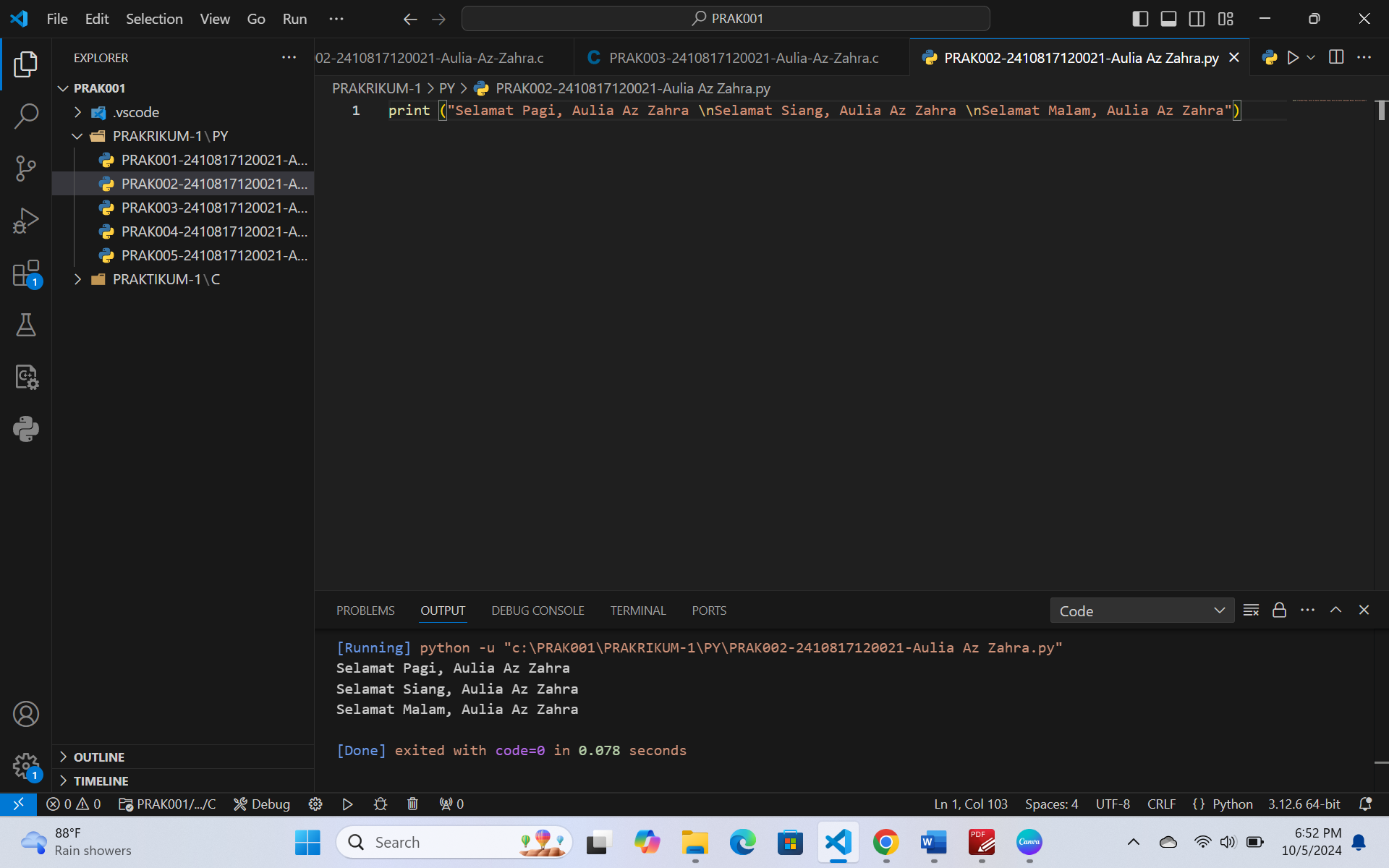
|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #include <stdio.h>  int main()  {      printf("Selamat Pagi, Aulia Az Zahra \nSelamat Siang, Aulia Az Zahra \nSelamat Malam, Aulia Az Zahra");  } |

Table 3 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2 | print ("Selamat Pagi, Aulia Az Zahra \nSelamat Siang, Aulia Az Zahra \nSelamat Malam, Aulia Az Zahra") |

Table 4 Source Code Bahasa Python Soal 2

## Sebuah gambar berisi teks, cuplikan layar, software, Software multimediaOutput Program



Gambar 3 Screenshot Output Bahasa Python Soal 2

Gambar 4 Screenshot Output Bahasa C Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda “ “ digunakan untuk mengapit teks yang akan ditampilkan. Fungsi ( ; ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Syntax (*\n*)atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan.

1. Pembahasan Bahasa Python

* Pada baris 1: syntax print() berfungsi untuk menampilkan output ke layer. Syntax (””) digunakan untuk mengapit teks yang akan ditampilkan. Pada python, untuk membuat baris baru dapat menggunakan *(\n)* untuk satu kolom yang sama. Contoh: ("Selamat Pagi, Aulia Az Zahra \nSelamat Siang, Aulia Az Zahra \nSelamat Malam, Aulia Az Zahra")

# SOAL 3

1. Buatlah program yang dapat menghasilkan output sebagaimana berikut:

|  |
| --- |
| **Output** |
| Andi Berkata “Saya Pasti Bisa” |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #include <stdio.h>  int main()  {      printf("Andi Berkata \"Saya Pasti Bisa\"");      return 0;  } |

Table 5 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1 | print ('Andi Berkata "Saya Pasti Bisa"') |

Table 6 Source Code Bahasa Python Soal 3

## Output ProgramSebuah gambar berisi teks, cuplikan layar, software, Software multimedia Deskripsi dibuat secara otomatisSebuah gambar berisi teks, cuplikan layar, software, Software multimedia Deskripsi dibuat secara otomatis

Gambar 5 Screenshot Output Bahasa C Soal 3

3

Gambar 6 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().* Syntax (\”) merupakan karakter spesial yang berfungsi untuk menyisipkan tanda kutip ganda dalam string. Sehingga pada output tanda (““) akan ditampilkan.
* Pada baris 4: syntax *printf (“ “);* adalah input yang akan dikeluarkan oleh output. Fungsi tanda (“ “) digunakan untuk mengapit teks yang akan ditampilkan. Fungsi ( ; ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan.
* Pada baris 5: syntax *return 0;* digunakan di akhir fungsi *main()* yang menandakan program telah selesai dijalankan dengan sukses.

1. Pembahasan Bahasa Python

* Pada baris 1: syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (””) digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 4

1. Buatlah program yang dapat menghasilkan output sebagaimana berikut:

|  |
| --- |
| **Output** |
| ##############################  # #  # Nama Anda #  # NIM #  # #  ############################## |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #include <stdio.h>  int main()  {      printf("###############################\n");      printf("#                             #\n");      printf("#        Aulia Az Zahra       #\n");      printf("#        2410817120021        #\n");      printf("#                             #\n");      printf("###############################\n");  } |

Table 7 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("##############################")  print("#                            #")  print("#        Aulia Az Zahra      #")  print("#        2410817120021       #")  print("#                            #")  print("##############################") |

Table 8 Source Code Bahasa Python Soal 4

## Output Program

Gambar 7 Screenshot Output Bahasa C Soal 4

3

Gambar 8 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: syntax printf(“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda “ “ digunakan untuk mengapit teks yang akan ditampilkan. Fungsi ( ; ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan.

1. Pembahasan Bahasa Python

* syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax *(””)* digunakan untuk mengapit teks yang akan ditampilkan. Pada python tidak menggunakan karakter spesial (*\n* ) untuk membuat baris baru.

# SOAL 5

1. Buatlah program yang dapat menampilkan huruf pertama dan terakhir nama anda dengan menggunakan tanda pagar (#). Misalnya, nama saya adalah Nadisheco, maka huruf yang harus dibuat adalah huruf **N** dan huruf **O** menggunakan tanda pagar (#). Perhatikan contoh output berikut:

|  |
| --- |
| **Output** |
| # # #######  ## # # #  ### # # #  # ## # # #  # ### # #  # ## # #  # # ####### |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #include <stdio.h>  int main()  {   printf("#####      ##    \n");   printf("#    #    #  #   \n");   printf("#    #   #    #  \n");   printf("#####    ######  \n");   printf("#   #    #    #  \n");   printf("#    #   #    #  \n");  } |

Table 9 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("#####      ##    ");  print("#    #    #  #   ");  print("#    #   #    #  ");  print("#####    ######  ");  print("#   #    #    #  ");  print("#   #   #    #  "); |

Table 10 Source Code Bahasa Python Soal 5

## Output Program

Gambar 9 Screenshot Output Bahasa C Soal 5

3

Gambar 10 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: syntax printf(“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda (“ “) digunakan untuk mengapit teks yang akan ditampilkan. Fungsi ( ; ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan.

1. Pembahasan Bahasa Python

* syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax *(””)* digunakan untuk mengapit teks yang akan ditampilkan. Pada python tidak menggunakan karakter spesial (*\n* ) untuk membuat baris baru.

# MODUL 1 : Variabel, Tipe Data dan Operator

# SOAL 1

1. Buatlah program yang dapat menghasilkan output dengan menampilkan variabel yang telah didefinisikan serta menampilkan hasil penjumlahan variabel tersebut dalam bahasa Python dan C.

|  |
| --- |
| **Output** |
| Variabel x bernilai 5  Variabel y bernilai 7  Variabel z bernilai 9  Jumlah variabel tersebut adalah 21 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | #include <stdio.h>  int main () {      int x = 5;      int y = 7;      int z = 9;      printf("Variabel x bernilai %d\n", x);      printf("Variabel y bernilai %d\n", y);      printf("Variabel z bernilai %d\n", z);      printf("Jumlah variabel tersebut adalah %d\n", x + y + z);      return 0;  } |

Table 11 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | x = 5  y = 7  z = 9  print ("Variabel x bernilai", x)  print ("Variabel y bernilai", y)  print ("Variabel z bernilai", z)  print ("Jumlah variabel tersebut adalah", x+y+z) |

Table 12 Source Code Bahasa Python Soal 1

## Output Program

Gambar 11 Screenshot Output Bahasa C Soal 1



Gambar 12 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C
   * Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
   * Pada baris 2-5: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().* Tiga variabel x, y, dan z dinyatakan dengan nilai 5, 7, dan 9 secara berturut-turut.
   * Pada baris 6-9: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda “ “ digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.
2. Pembahasan Bahasa Python

* Pada baris 1-3: Tiga variabel x, y, dan z dinyatakan dengan nilai 5, 7, dan 9 secara berturut-turut.
* Pada baris 4-7: syntax print() berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 2

2. Buatlah program yang dapat menghasilkan output berupa hasil kombinasi perkalian dan pembagian dari variabel yang telah didefinisikan.

|  |
| --- |
| **Output** |
| Variabel a bernilai 4  Variabel b bernilai 8  Variabel c bernilai 3  Hasil dari a dikali b dibagi c adalah 10.666667 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | #include <stdio.h>  int main()  {      float a, b, c;      a = 4;      b = 8;      c = 3;      printf("Variabel a bernilai %.f\n", a);      printf("Variabel b bernilai %.f\n", b);      printf("Variabel c bernilai %.f\n", c);      printf("Hasil dari a dikali b dibagi c adalah %6f", a \* b / c);      return 0;  } |

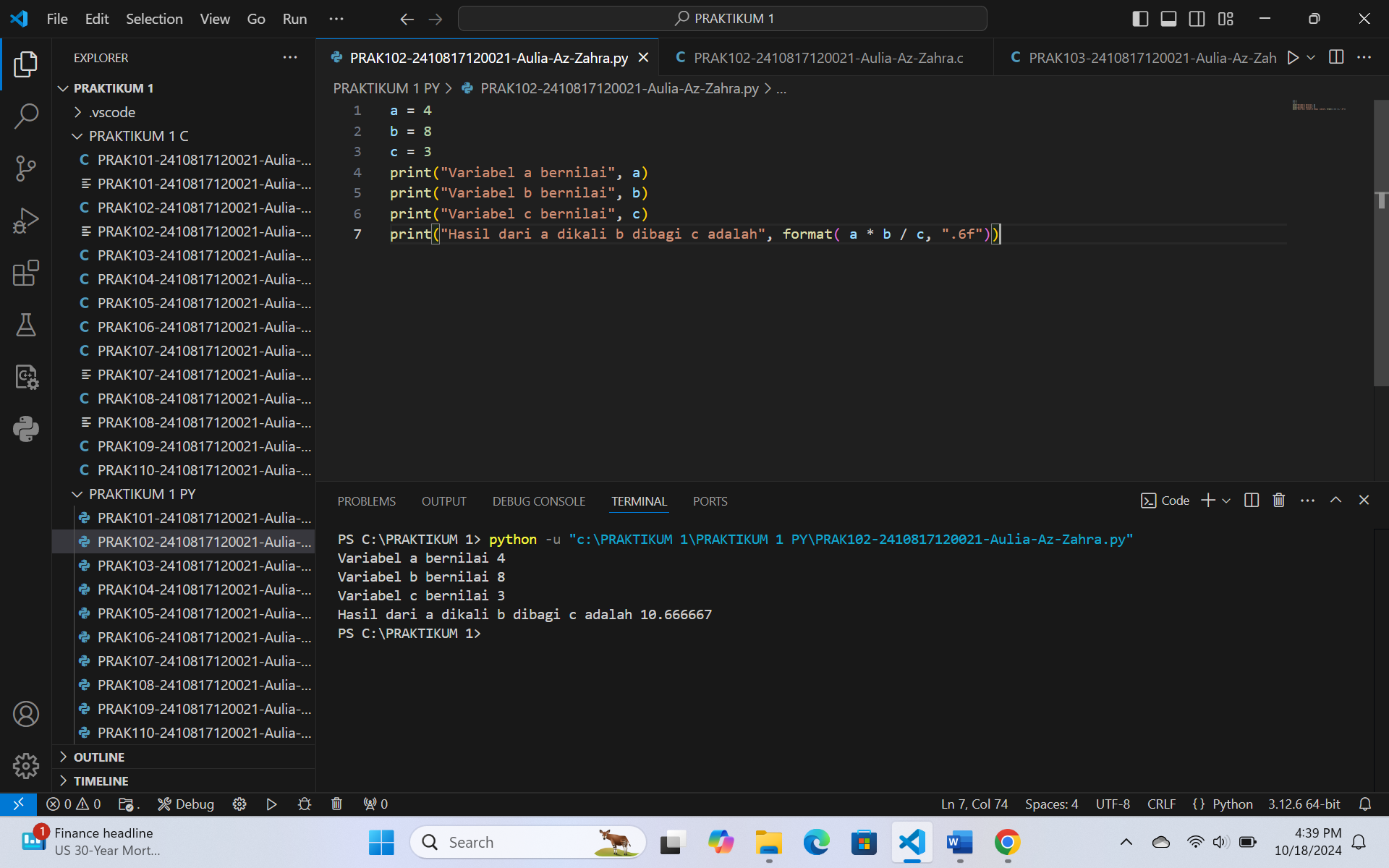
Table 13 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | a = 4  b = 8  c = 3  print("Variabel a bernilai", a)  print("Variabel b bernilai", b)  print("Variabel c bernilai", c)  print("Hasil dari a dikali b dibagi c adalah", format( a \* b / c, ".6f")) |

Table 14 Source Code Bahasa Python Soal 2

## Output Program

Gambar 13 Screenshot Output Bahasa C Soal 2

****

Gambar 14 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-7: Tiga variabel a, b, dan c dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. Variabel diinisialisasi dengan nilai 4, 8, dan 3 secara berturut-turut.
* Pada baris 8-11: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. format *(%6f,)* berarti hasil akan ditampilkan sebagai angka desimal dengan enam angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-3: Tiga variabel a, b, dan c dideklarasikan dan diinisialisasi dengan nilai 4, 8, dan 3 secara berturut-turut.
* Pada baris 4-7: Pada baris 4-7: syntax print() berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. Format *(.6f,)* berarti hasil akan ditampilkan sebagai angka desimal dengan enam angka setelah koma.

# SOAL 3

3. Buatlah program yang dapat menghasilkan output berupa hasil kombinasi pertambahan, pengurangan, perkalian dan pembagian dari variabel yang telah didefinisikan.

|  |
| --- |
| **Output** |
| Variabel a bernilai 9  Variabel b bernilai 6  Variabel x bernilai 10  Variabel y bernilai 7  Hasil dari a ditambah b dikali x dan dibagi y adalah 21.43 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | #include <stdio.h>  int main()  {      float a, b, x, y;      a = 9;      b = 6;      x = 10;      y = 7;      printf("variabel a bernilai %.f\n", a);      printf("Variabel b bernilai %.f\n", b);      printf("Variabel x bernilai %.f\n", x);      printf("Variabel y bernilai %.f\n", y);      printf("Hasil dari a ditambah b dikali x dan dibagi y adalah %.2f", (a + b) \* x / y);      return 0;  } |

Table 15 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | a = 9  b = 6  x = 10  y = 7  print("Variabel a bernilai", a)  print("Variabel b bernilai", b)  print("Variabel x bernilai", x)  print("Variabel y bernilai", y)  print("Hasil dari a ditambah b dikali x dan dibagi y adalah", format ((a + b) \* x / y, ".2f")) |

Table 16 Source Code Bahasa Python Soal 3

## Output Program

Gambar 15 Screenshot Output Bahasa C Soal 3

3

Gambar 16 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud (int) di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-8: Empat variabel a, b, x, dan y dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. Variabel diinisialisasi dengan nilai 9, 6, 10 dan 7 secara berturut-turut.
* Pada baris 9-13: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. format *(%.2f,)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-4: Empat variabel a, b, x, dan y dideklarasikan dan diinisialisasi dengan nilai 9, 6, 10, dan 7 secara berturut-turut.
* Pada baris 5-9: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. Format *(.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma.

# SOAL 4

4. Buatlah program yang dapat menghasilkan output berupa hasil diskon dari sebuah harga dengan mengimplementasikan operator.

|  |
| --- |
| **Output** |
| Harga sepatu A adalah 400000  Harga sepatu B adalah 350000  Sepatu A mendapat diskon 13% sehingga harganya menjadi 348000  Sepatu B mendapat diskon 21% sehingga harganya menjadi 276500 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | #include <stdio.h>  int main()  {      int sepatu\_a = 400000;      int sepatu\_b = 350000;      printf("Harga sepatu A adalah %d\n", sepatu\_a);      printf("Harga sepatu B adalah %d\n", sepatu\_b);      printf("Sepatu A mendapatkan diskon 13%% sehingga harganya menjadi %d\n", sepatu\_a - (sepatu\_a \* 13/100));      printf("Sepatu B mendapatkan diskon 21%% sehingga harganya menjadi %d\n", sepatu\_a - (sepatu\_a \* 21/100));      return 0;  } |

Table 17 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | sepatu\_A = 400000  sepatu\_B = 350000  harga\_diskonA = sepatu\_A - (sepatu\_A \* 13/100)  harga\_diskonB = sepatu\_B - (sepatu\_B \* 21/100)  print("Harga sepatu A adalah", sepatu\_A)  print("Harga sepatu B adalah", sepatu\_B)  print("Sepatu A mendapatkan diskon 13% sehingga harganya menjadi", int(harga\_diskonA))  print("Sepatu B mendapatkan diskon 21% sehingga harganya menjadi", int(harga\_diskonB)) |

Table 18 Source Code Bahasa Python Soal 4

## Output Program

Gambar 17 Screenshot Output Bahasa C Soal 4

3

Gambar 18 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-5: Dua variabel, sepatu\_a dan sepatu\_b, dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan harga masing-masing sepatu, yaitu 400.000 dan 350.000.
* Pada baris 6-9: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. % dalam string perlu diulang *(%%)* agar ditampilkan sebagai simbol persen. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-2: Dua variabel, sepatu\_a dan sepatu\_b, dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan harga masing-masing sepatu, yaitu 400.000 dan 350.000.
* Pada baris 6-7: syntax print() berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. Dalam python % dalam string tidak perlu diulang *(%%)* agar ditampilkan sebagai simbol persen.

# SOAL 5

5. Buatlah program yang dapat menghasilkan output berupa total sisa bagi dari pembagian dengan mengimplementasikan operator.

|  |
| --- |
| **Output** |
| Variabel a bernilai 9  Variabel b bernilai 5  Variabel x bernilai 8  Variabel y bernilai 8  Total sisa bagi dari a dibagi b dan x dibagi y adalah 4 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | #include <stdio.h>  int main()  {      int a, b, x, y;      a = 9, b = 5, x = 8, y = 8;      int modulus\_ab = a % b;      int modulus\_xy = x % y;      printf("Variabel a bernilai %d\n", a);      printf("Variabel b bernilai %d\n", b);      printf("Variabel x bernilai %d\n", x);      printf("Variabel y bernilai %d\n", y);      printf("Total sisa bagi dari a dibagi b dan x dibagi y adalah %d", modulus\_ab + modulus\_xy);      return 0;  } |

Table 19 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | a = 9  b = 5  x = 8  y = 8  modulus\_ab = a % b  modulus\_xy = x % y  print("Variabel a bernilai", a)  print("Variabel b bernilai", b)  print("Variabel x bernilai", x)  print("Variabel y bernilai", y)  print("total sisa bagi dari a dibagi b dan x dibagi y adalah", modulus\_ab + modulus\_xy) |

Table 20 Source Code Bahasa Python Soal 5

## Output Program

Gambar 19 Screenshot Output Bahasa C Soal 5

3

Gambar 20 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-5: Empat variable a, b x, dan y dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan nilai 9, 5, 8, dan 8 berturut-turut.
* Pada baris 6: *(modulus\_ab = a % b)* akan menghasilkan sisa bagi dari 9 % 5.
* Pada baris 7: *(modulus\_xy = x % y)* akan menghasilkan sisa bagi dari 8 % 8.
  + Pada baris 8-12: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-4: Empat variabel a, b, x, dan y dideklarasikan dan diinisialisasi dengan nilai 9, 5, 8, dan 8 secara berturut-turut.
* Pada baris 5: *(modulus\_ab = a % b)* akan menghasilkan sisa bagi dari 9 % 5.
* Pada baris 6: *(modulus\_xy = x % y)* akan menghasilkan sisa bagi dari 8 % 8.
* Pada baris 8-12: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 6

6. Buatlah program yang dapat menghasilkan output berupa hasil pengecekan antara 2 nilai.

|  |
| --- |
| **Output** |
| Variabel a bernilai 4  Variabel b bernilai 8  Variabel c bernilai 3  Apakah a sama dengan b ? jawabannya adalah 0  Apakah b lebih besar dari c ? jawabannya adalah 1  Apakah a tidak sama dengan c ? jawabannya adalah 1 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | #include <stdio.h>  int main()  {      int a, b, c;      a = 4, b = 8, c = 3;      printf("Variabel a bernilai %d\n", a);      printf("Variabel b bernilai %d\n", b);      printf("Variabel c bernilai %d\n", c);      printf("Apakah a sama dengan b? jawabannya adalah %d\n", a == b);      printf("Apakah b lebih besar dari c? jawabannya adalah %d\n", b > c);      printf("Apakah a tidak sama dengan c? jawabannya adalah %d\n", a != c);      return 0;  } |

Table 21 Source Code Bahasa C Soal 6

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | a = 4  b = 8  c = 3  print("Variabel a bernilai ", a)  print("Variabel b bernilai ", b)  print("Variabel c bernilai ", c)  print("Apakah a sama dengan b? jawabannya adalah", int(a == b))  print("Apakah b lebih besar dari c? jawabannya adalah", int(b > c))  print("Apakah a tidak sama dengan c? jawabannya adalah", int(a != c)) |

Table 22 Source Code Bahasa Python Soal 6

## Output Program

Gambar 21 Screenshot Output Bahasa C Soal 6

3

Gambar 22 Screenshot Output Bahasa Python Soal 6

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-5: Tiga variable a, b, dan c dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan nilai 4, 8, dan 3 berturut-turut.
  + Pada baris 6-11: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python
   * Pada baris 1-3: Tiga variabel a, b, dan c dideklarasikan dan diinisialisasi dengan nilai 4, 8, dan 3 secara berturut-turut.
   * Pada baris 4-9: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 7

7. Pak Dengklek baru saja membeli sebidang tanah berbentuk segitiga sebarang dengan panjang tiap sisi tanah berturut-turut 4 m, 5 m, dan 7 m. Tanah tersebut hendak dipasang pagar agar bebek yang di pelihara Pak Dengklek tidak berkeliaran sembarangan. Biaya pemasangan pagar per meter-nya adalah Rp 85.000,00.

Buatkan sebuah program untuk membantu Pak Dengklek menghitung biaya yang diperlukan agar seluruh tanahnya dikelilingi pagar.

|  |
| --- |
| **Output** |
| Diketahui :  Panjang sisi segitiga berturut-turut adalah 4, 5, dan 7  Keliling Tanah Pak Dengklek adalah 16  Harga tanah Per Meter adalah 85000  Jawaban :  Biaya yang diperlukan Pak Dengklek adalah : Rp 1360000 |

## 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 | #include <stdio.h>  int main()  {      int sisi1, sisi2, sisi3;      sisi1 = 4, sisi2 = 5, sisi3 = 7;      int biaya\_per\_meter = 85000;      int keliling = sisi1 + sisi2 + sisi3;      int biaya\_total = keliling \* biaya\_per\_meter;      printf("Diketahui :\n");      printf("Panjang sisi segitiga berturut-turut adalah %d, ", sisi1);      printf("%d, ", sisi2);      printf("%d\n", sisi3);      printf("Keliling Tanah Pak Dengklek adalah %d\n", keliling);      printf("Harga tanah Per Meter adalah %d\n", biaya\_per\_meter);      printf("Jawaban :\n");      printf("Biaya yang diperlukan Pak Dengklek adalah : Rp %d", biaya\_total);      return 0;} |

Table 23 Source Code Bahasa C Soal 7

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | sisi1 = 4  sisi2 = 5  sisi3 = 7  biaya\_per\_meter = 85000  keliling = sisi1 + sisi2 + sisi3  biaya\_total = keliling \* biaya\_per\_meter  print("Diketahui :")  print("Panjang sisi segitiga berturut-turt adalah %d, %d, %d" % (sisi1, sisi2, sisi3))  print("Keliling Tanah Pak Dengklek adalah %d" % keliling)  print("Harga tanah Per Meter adalah %d" % biaya\_per\_meter)  print("Jawaban :")  print("Biaya yang diperlukan Pak Dengklek adalah : Rp %d" % biaya\_total) |

Table 24 Source Code Bahasa Python Soal

## Output Program

Gambar 23 Screenshot Output Bahasa C Soal 7

3

Gambar 24 Screenshot Output Bahasa Python Soal 7

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud (*int*) di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().*
* Pada baris 4-5: Tiga variabel *sisi1, sisi2,* dan *sisi3* dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan nilai 4, 5, dan 7 berturut-turut.
* Pada baris 7:(*Biaya\_per\_meter*) menyimpan nilai 85000, yang merepresentasikan biaya per meter tanah.
* Pada baris 8: (Keliling) dihitung sebagai jumlah dari ketiga sisi segitiga (*sisi1 + sisi2 + sisi3).*
* Pada baris 9: (*Biaya\_total*) dihitung sebagai hasil kali dari keliling dan biaya\_per\_meter
  + Pada baris 11-18: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python
   * Pada baris 1-3: Tiga variabel sisi1, sisi2, dan sisi3 dideklarasikan dan diinisialisasi dengan nilai 4, 5, dan 7 secara berturut-turut.

* Pada baris 4:(*Biaya\_per\_meter*) menyimpan nilai 85000, yang merepresentasikan biaya per meter tanah.
* Pada baris 5: (Keliling) dihitung sebagai jumlah dari ketiga sisi segitiga (*sisi1 + sisi2 + sisi3).*
* Pada baris 6: (*Biaya\_total*) dihitung sebagai hasil kali dari keliling dan biaya\_per\_meter
* Pada baris 8-13: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat.

# SOAL 8

8. Hari ini Pak Dengklek jogging mengelilingi taman berbentuk lingkaran sebanyak 5 putaran. Berdasarkan aplikasi Runkeeper pada smartphone yang digunakan, Pak Dengklek telah berlari sejauh 14 kilometer. Berapakah jari-jari taman yang dikelilingi Pak Dengklek?

|  |
| --- |
| **Output** |
| Diketahui : Pak Dengklek mengelilingi taman = 5 Putaran  Jarak tempuh Pak Dengklek = 14 Kilometer  Jawaban :  Jari-jari taman yang dikelilingi Pak Dengklek adalah 0.45 Kilometer |

## 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()  {      float putaran = 5;      float jarak\_total = 14;      float jari\_jari = jarak\_total / (2 \* 3.14 \* 5);      printf("Diketahui :\n");      printf("Pak Dengklek mengelilingi taman = %.f Putaran\n", putaran);      printf("Jarak tempuh Pak Dengklek = %.f kilometer\n", jarak\_total);      printf("\n");      printf("Jawaban :\n");      printf("Jari-jari taman yang dikelilingi Pak Dengklek adalah %.2f Kilometer", jari\_jari);      return 0;  } |

Table 25 Source Code Bahasa C Soal 8

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | putaran = 5  jarak = 14  jarijari = jarak / (2 \* 3.14 \* 5)  print("Diketahui :")  print("Pak Dengklek mengelilingi taman = %.f Putaran" % (putaran))  print("Jarak tempuh Pak Dengklek = %.f Kilometer" % (jarak))  print()  print("Jawaban :")  print("Jari-jari taman yang dikelilingi Pak Dengklek adalah %.2f Kilometer" % (jarijari)) |

Table 26 Source Code Bahasa Python Soal 8

## Output Program

Gambar 25 Screenshot Output Bahasa C Soal 8

3

Gambar 26 Screenshot Output Bahasa Python Soal 8

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4-6: Tiga variabel *putaran, jarak\_total,* dan *jari\_jari* dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. Variabel diinisialisasi dengan nilai 5, 14, dan *jari-jari = jarak\_total / (2 \* 3,14 \* 5)* secara berturut-turut.
* Pada baris 8-11: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. Format *(%.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-3: Tiga variabel *putaran, jarak\_total,* dan *jari\_jari*, variabel diinisialisasi dengan nilai 5, 14, dan *jari-jari = jarak\_total / (2 \* 3,14 \* 5)* secara berturut-turut.
* Pada baris 5-10: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Format *(%.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma.

# SOAL 9

9. Cardia Riverlands merupakan tanah yang damai sampai Yu Zhong jelmaan dari Black Dragon datang dengan membawa 958.730 pasukan dan memporak-porandakan tempat tersebut. Mendengar berita atas kacaunya Cardia Riverlands, padepokan Dragon Altar mengirim pahlawan yang terdiri dari Zilong, Ling, Baxia, Wanwan, dan Chang’e dengan misi untuk mengalahkan semua pasukan Yu Zhong. Jika para pahlawan Dragon Altar bersepakat untuk membagi musuh sama rata, berapa pasukan yang harus dikalahkan oleh setiap pahlawan Dragon Altar?

Buatlah program untuk menghitung jumlah pasukan yang harus dikalahkan oleh setiap pahlawan Dragon Altar!

|  |
| --- |
| **Output** |
| Jumlah pasukan yang dibawa Yu Zhong = ?  Jumlah pahlawan = ?  Jumlah pasukan yang harus dikalahkan setiap pahlawan adalah 191746 pasukan |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | #include <stdio.h>  int main()  {      int pasukan\_yuzhong = 958730;      int pahlawan = 5;      int pasukan\_per\_pahlawan = pasukan\_yuzhong / pahlawan;      printf("Jumlah pasukan yang dibawa Yu Zhong = %d\n", pasukan\_yuzhong);      printf("Jumlah pahlawan = %d\n", pahlawan);      printf("Jumlah pasukan yang harus dikalahkan setiap pahlawan adalah %d pasukan", pasukan\_per\_pahlawan);      return 0;  } |

Table 27 Source Code Bahasa C Soal 9

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | pasukan\_yuzhong = 958730  pahlawan = 5  pasukan\_per\_pahlawan = pasukan\_yuzhong / pahlawan  print("Jumlah pasukan yang dibawa Yu Zhong = %d" % (pasukan\_yuzhong))  print("Jumlah pahlawan = %d" % (pahlawan))  print("Jumlah pasukan yang harus dikalahkan setiap pahlawan adalah %d" % (pasukan\_per\_pahlawan)) |

Table 28 Source Code Bahasa Python Soal 9

## Output Program

Gambar 27 Screenshot Output Bahasa C Soal 9

3

Gambar 28 Screenshot Output Bahasa Python Soal 9

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud (*int*) di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().*
* Pada baris 4-6: Tiga variabel *pasukan\_yuzhong, pahlawan,* dan *pasukan\_per\_pahlawan* dideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan nilai 958730, 5, dan *pasukan\_yuzhong / pahlawan* berturut-turut.
  + Pada baris 8-10: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-3: Tiga variabel *pasukan\_yuzhong, pahlawan,* dan *pasukan\_per\_pahlawan.* Diinisialisasi dengan nilai 958730, 5, dan *pasukan\_per\_pahlawan =* *pasukan\_yuzhong / pahlawan* berturut-turut.
* Pada baris 5-7: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat.

# SOAL 10

10. Buatlah sebuah program yang digunakan untuk menghitung keliling dan luas segitiga sikusiku jika diketahui alas = 5cm dan tinggi = 12cm.



|  |
| --- |
| **Output** |
| Diketahui :  Alas = 5 cm  Tinggi = 12 cm  Jawab : S  isi A = ? cm  Sisi B = ? cm  Sisi C = ? cm  Keliling = 30 cm  Luas = 30 cm |

## 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 | # #include <stdio.h>  #include <math.h>  int main ()  {      int alas, tinggi, keliling, luas;      alas = 5;      tinggi = 12;      keliling = 30;      luas = 30;      int miring = sqrt(alas\*alas + tinggi\*tinggi);      printf("Diketahui :\n");      printf("Alas = %d cm\n", alas);      printf("Tinggi = %d cm\n", tinggi);      printf("\n");      printf("Jawab :\n");      printf("Sisi A = %d cm\n", tinggi);      printf("Sisi B = %d cm\n", miring);      printf("Sisi C = %d cm\n", alas);      printf("Keliling = %d cm\n", keliling);      printf("Luas = %d cm\n", luas);      return 0;  } |

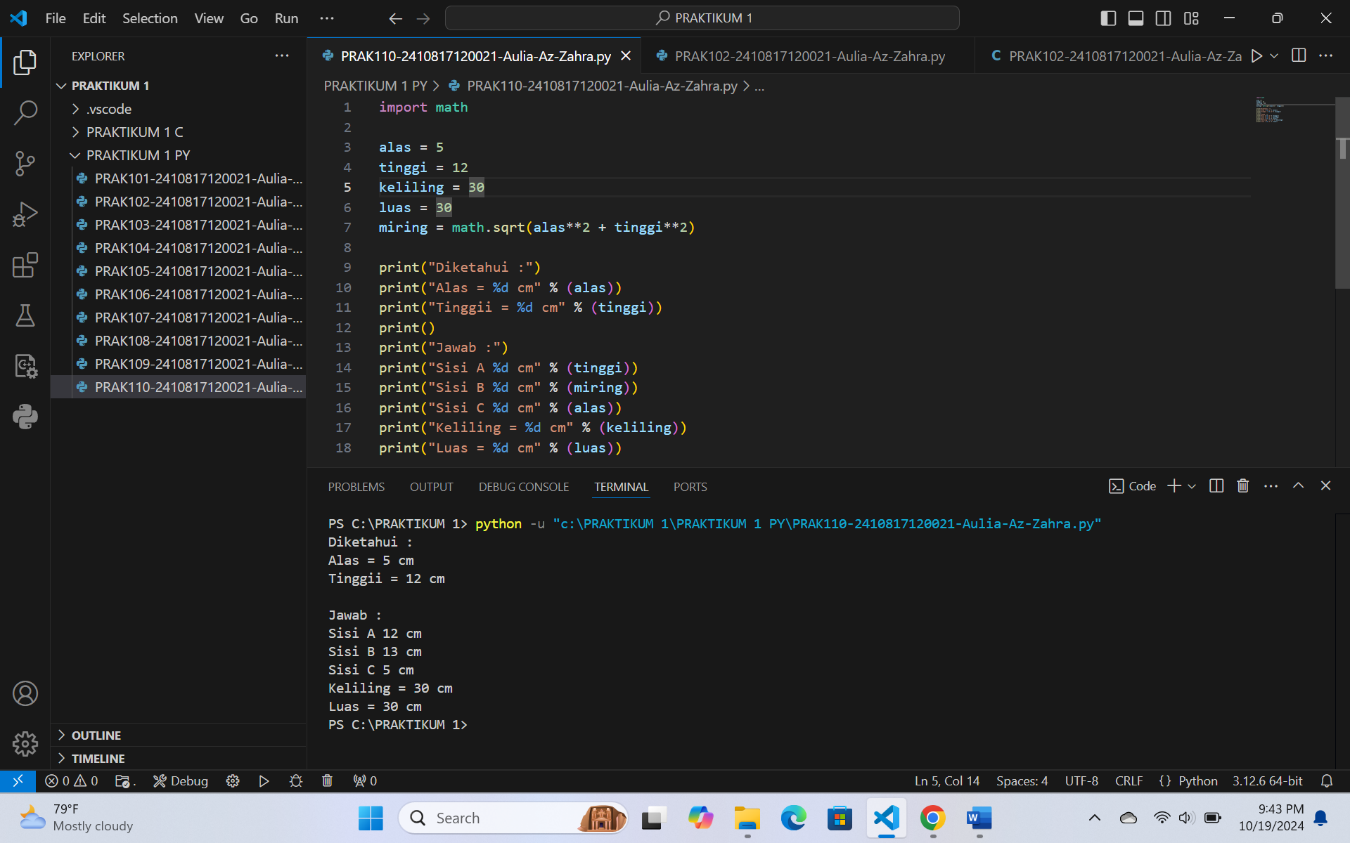
Table 29 Source Code Bahasa C Soal 10

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | import math  alas = 5  tinggi = 12  keliling = 30  luas = 30  miring = math.sqrt(alas\*\*2 + tinggi\*\*2)  print("Diketahui :")  print("Alas = %d cm" % (alas))  print("Tinggii = %d cm" % (tinggi))  print()  print("Jawab :")  print("Sisi A %d cm" % (tinggi))  print("Sisi B %d cm" % (miring))  print("Sisi C %d cm" % (alas))  print("Keliling = %d cm" % (keliling))  print("Luas = %d cm" % (luas)) |

Table 30 Source Code Bahasa Python Soal 10

## Output Program

Gambar 29 Screenshot Output Bahasa C Soal 10



Gambar 30 Screenshot Output Bahasa Python Soal 10

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 2: #include <math.h> digunakan untuk fungsi matematika seperti sqrt yang menghitung akar kuadrat.
  + Pada baris 6-10: Empat variabel *alas, tinggi,* keliling, dan luasdideklarasikan sebagai tipe data (*int*) dan diinisialisasi dengan nilai 5, 12, 30 dan 30 berturut-turut.
  + Pada baris 11: int miring = sqrt (alas\*alas + tinggi\*tinggi); menggunakan rumus Pythagoras untuk menghitung panjang sisi miring segitiga. Fungsi *(sqrt)* menghitung akar kuadrat dari hasil penjumlahan kuadrat alas dan kuadrat tinggi.
  + Pada baris 13-22: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax (*\n*) atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi (*return 0;* ) menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Import math library ini diimpor untuk menggunakan fungsi matematika, khususnya math.sqrt untuk menghitung akar kuadrat.
* Pada baris 2-5: Empat variabel *alas, tinggi,* keliling, dan luas. Diinisialisasi dengan nilai 5, 12, 30 dan 30 berturut-turut.
* Pada baris 6: Miring = *math.sqrt (alas\*\*2 + tinggi\*\*2)* menggunakan rumus Pythagoras untuk menghitung panjang sisi miring dari segitiga siku-siku, di mana miring adalah hasil dari akar kuadrat (math.sqrt) dari jumlah kuadrat alas dan tinggi.
* Pada baris 8-17: Syntax *print()* berfungsi untuk menampilkan output ke layer. Syntax (*””*) digunakan untuk mengapit teks yang akan ditampilkan. (*%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat.

# MODUL 2 : Input dan Output

# SOAL 1

1. Buatlah program yang dapat menginput biodata dan menghasilkan output dengan menampilkan biodata tersebut dalam bahasa Python dan C.

|  |  |
| --- | --- |
| **Input** |  |
| Nama | : |
| NIM | : |
| Kelas Paralel | : |
| Tempat/Tanggal Lahir | : |
| Alamat | : |
| Hobby | : |
| No. HP | : |
| **Output** |  |
| Nama | : |
| NIM | : |
| Kelas Paralel | : |
| Tempat/Tanggal Lahir | : |
| Alamat | : |
| Hobby | : |
| No. HP | : |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  2  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | #include <stdio.h>  int main ()  {      char nama[100], nim[100], kelas[100], ttl[100],      alamat[100], hobby[100], no\_hp[100];      printf("Nama                 : ");      fgets(nama, sizeof(nama), stdin);      printf("NIM                  : ");      fgets(nim, sizeof(nim), stdin);      printf("Kelas Paralel        : ");      fgets(kelas, sizeof(kelas), stdin);      printf("Tempat/Tanggal Lahir : ");      fgets(ttl, sizeof(ttl), stdin);      printf("Alamat               : ");      fgets(alamat, sizeof(alamat), stdin);      printf("Hobby                : ");      fgets(hobby, sizeof(hobby), stdin);      printf("No. HP               : ");      fgets(no\_hp, sizeof(no\_hp), stdin);      printf("\n");      printf("Nama                 : %s", nama);      printf("NIM                  : %s", nim);      printf("Kelas Paralel        : %s", kelas);      printf("Tempat/Tanggal Lahir : %s", ttl);      printf("Alamat               : %s", alamat);      printf("Hobby                : %s", hobby);      printf("No. HP               : %s", no\_hp);      return 0;  } |

Table 31 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | nama   = input("Nama                   : ")  nim    = input("NIM                    : ")  kelas  = input("Kelas Paralel          : ")  ttl    = input("Tempat/Tanggal Lahir   : ")  alamat = input("Alamat                 : ")  hobby  = input("Hobby                  : ")  no\_hp  = input("No. HP                 : ")  print("Nama                   : " + nama)  print("NIM                    : " + nim)  print("Kelas Paralel          : " + kelas)  print("Tempat/Tanggal Lahir   : " + ttl)  print("Alamat                 : " + alamat)  print("Hobby                  : " + hobby)  print("No. HP                 : " + no\_hp) |

Table 32 Source Code Bahasa Python Soal 1

## Output ProgramSebuah gambar berisi teks, cuplikan layar, software, Software multimedia Deskripsi dibuat secara otomatis

Gambar 31 Screenshot Output Bahasa C Soal 1

Gambar 32 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C
   1. Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
   2. Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi *main().*
   3. Pada baris 4-5: Array *(char [100])* untuk setiap data input (nama, nim, kelas, dll.) yang memiliki kapasitas 100 karakter
   4. Pada baris 6-28: Fungsi *fgets()* digunakan untuk membaca input. *Fgets()* juga memungkinkan pengguna memasukkan spasi dalam input. Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *“ “* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)*atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%s)* digunakan untuk menampilkan string yang diambil dari array (*char)*. Fungsi *( ; )* memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0;)* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.
2. Pembahasan Bahasa Python

* Pada baris 9-15: Syntax print() berfungsi untuk menampilkan output ke layer. Syntax *(””)* digunakan untuk mengapit teks yang akan ditampilkan.
* Pada baris 1-7: Fungsi *input()*menampilkan variabel yang disisipkan dari input. Input yang dimasukkan secara otomatis dianggap sebagai string, sehingga tidak perlu ada spesifikasi tipe data.

# SOAL 2

1. Buatlah program yang dapat menginputkan dan menghasilkan output berupa hasil dari nilai pertama ditambah nilai kedua adalah 34.50

**Test case ke 1 :**

|  |
| --- |
| **Input** |
| Masukkan Nilai Pertama : 14  Masukkan Nilai Kedua : 20.5 |
| **Output** |
| Hasil dari penjumlahan nilai pertama “14” dan nilai kedua “20.5” adalah “34.50” |

**Test case ke 2 :**

|  |
| --- |
| **Input** |
| Masukkan Nilai Pertama : 0.45  Masukkan Nilai Kedua : 99.5 |
| **Output** |
| Hasil dari penjumlahan nilai pertama “0.45” dan nilai kedua “99.5” adalah “99.95” |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | #include <stdio.h>  int main()  {      float nilaiPertama, nilaiKedua;      printf("Masukan Nilai Pertama   : ");      scanf("%g", &nilaiPertama);      printf("Masukan Nilai Kedua     : ");      scanf("%g", &nilaiKedua);      printf("Hasil dari penjumlahan nilai pertama \"%g\" dan nilai kedua \"%g\" adalah \"%.2f\"\n", nilaiPertama, nilaiKedua, nilaiPertama + nilaiKedua);      return 0;  } |

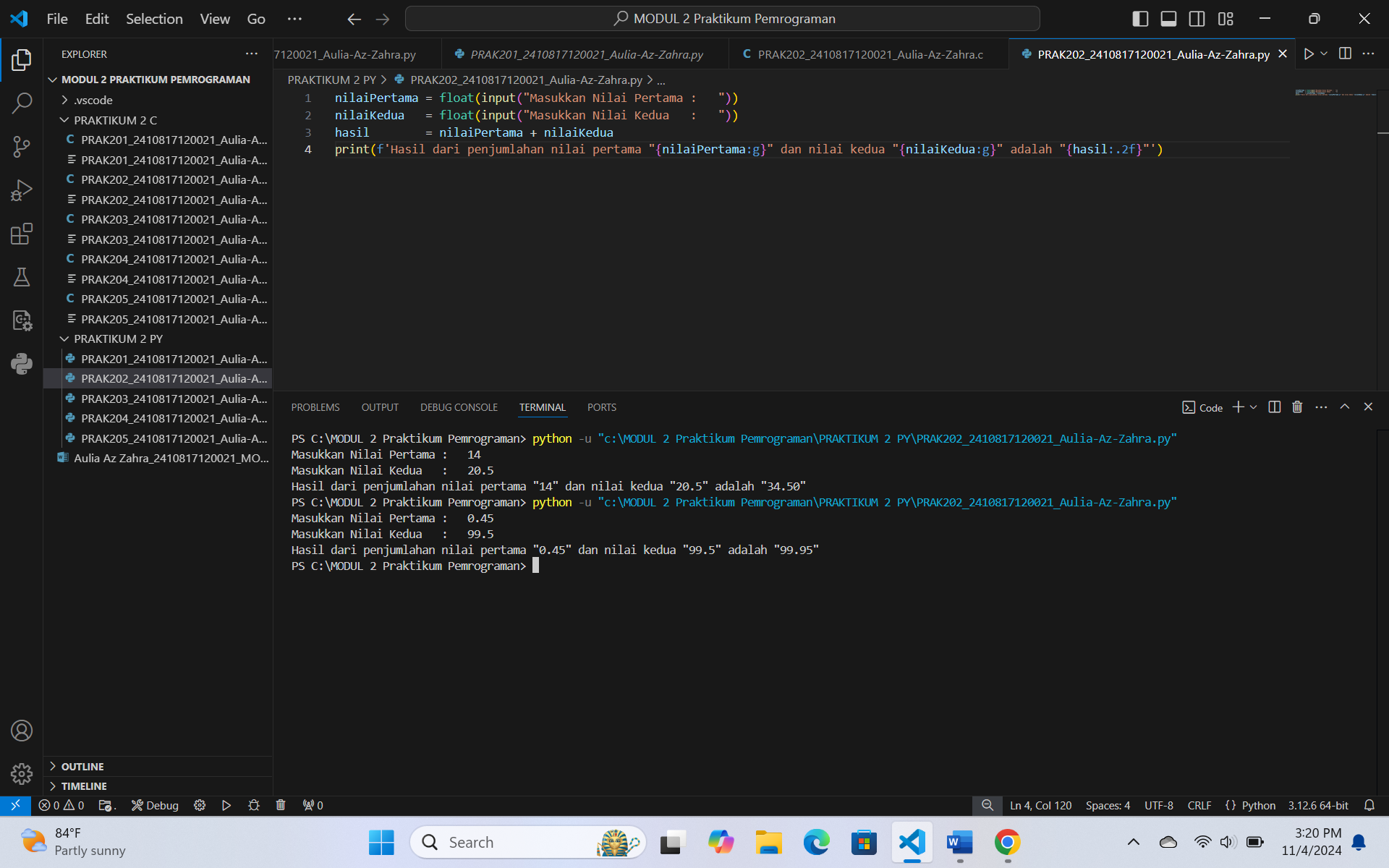
Table 33 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | nilaiPertama = float(input("Masukkan Nilai Pertama :   "))  nilaiKedua   = float(input("Masukkan Nilai Kedua   :   "))  hasil        = nilaiPertama + nilaiKedua  print(f'Hasil dari penjumlahan nilai pertama "{nilaiPertama:g}" dan nilai kedua "{nilaiKedua:g}" adalah "{hasil:.2f}"') |

Table 34 Source Code Bahasa Python Soal 2

## Output Program

Gambar 33 Screenshot Output Bahasa C Soal 2

****

Gambar 34 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 3: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 5: Dua variabel *nilaiPertama & nilaiKedua* diinisiasikan sebagai tipe data (*float)* untuk menyimpan angka desimal.
* Pada baris 7-12: syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *scanf()* digunakan untuk mengambil input. Format *(%g)* digunakan untuk membaca nilai float dari input. Format *(%.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-2: Dua variabel *nilaiPertama & nilaiKedua* dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. Fungsi *input()* menerima input dari pengguna.
* Pada baris 4: Syntax *(f’ ’)* digunakan untuk menyisipkan variabel langsung ke dalam string. Format *(%g)* digunakan untuk membaca nilai float dari input. Format *(%.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma.

# SOAL 3

1. Buatlah program yang dapat menghitung dan menghasilkan output langsung menampilkan berupa hasildari a dikurang b dikali dengan i dibagi j dikurang dengan x ditambah y adalah

|  |  |
| --- | --- |
| **Input** | **Output** |
| 20 3 4 12 5 9 | -8.333 |
| 12 2  10 4  3 14 | 8.000 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | #include <stdio.h>  int main()  {      float a, b, i, j, x, y;      scanf("%f", &a);      scanf("%f", &b);      scanf("%f", &i);      scanf("%f", &j);      scanf("%f", &x);      scanf("%f", &y);      printf("%.3f", (a - b) \* (i / j) - (x + y));      return 0;  } |

Table 35 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | a = input().split()  if len (a) == 6:      a, b, i, j, x, y = map(float, a)  elif len (a) == 2:      a, b = map(float, a)      i, j = map(float, input().split())      x, y = map(float, input().split())  hasil = (a - b) \* (i / j) - (x + y)  print(f"{hasil:.3f}") |

Table 36 Source Code Bahasa Python Soal 3

## Output Program

Gambar 35 Screenshot Output Bahasa C Soal 3

3

Gambar 36 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 3: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 5: Enam variabel a, b, i, j, x dan y dinyatakana sebagai tipe data (*float)* untuk menyimpan angka desimal.
* Pada baris 6-13: Fungsi *scanf()* digunakan untuk mengambil input. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Format *(%.3f)* berarti hasil akan ditampilkan sebagai angka desimal dengan tiga angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: *input()* menerima input sebagai satu string. *Split()* memecah string tersebut berdasarkan spasi.
* Pada baris 3-8: Fungsi *len(a)* digunakan untuk mendapatkan jumlah variabel dalam list a. Enam variabel a, b, i, j, x dan y diinisiasi sebagai tipe data (*float)* untuk menyimpan angka desimal. Fungsi *map()* digunakan untuk mengonversi setiap variabel dari list a ke dalam tipe data float.
* Pada baris 10: Syntax *(f’ ’)* digunakan untuk menyisipkan variabel langsung ke dalam string. Format *(.3f)* berarti hasil akan ditampilkan sebagai angka desimal dengan tiga angka setelah koma.

# SOAL 4

1. Hari ini pak dengklek berencana membeli sebuah bejana memakai tutup dan berbentuk tabung di pasar sukagadai. Di pasar, pak Dengklek menemukan banyak bejana yang membuatnya bingung bejana mana yang yang harus dibeli oleh Pak Dengklek. Buatlah program untuk mengetahui volume, luas, dan keliling bejana jika yang diketahui hanya jari- jari dan tinggi bejana tersebut.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 7  10 | Volume = 1540.00  Luas = 748.00  Keliling = 44.00 |
| 10 10 | Volume = 3142.86  Luas = 1257.14  Keliling = 62.86 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  22 | #include <stdio.h>  int main()  {      float jarijari, tinggi;      float phi = 22.0 / 7.0;      scanf("%f", &jarijari);      scanf("%f", &tinggi);        float volume   = phi \* jarijari \* jarijari \* tinggi;      float luas     = 2 \* phi \* jarijari \* tinggi + 2 \* phi \* jarijari \* jarijari;      float keliling = 2 \* phi \* jarijari;      printf("Volume   = %.2f\n", volume);      printf("Luas     = %.2f\n", luas);      printf("Keliling = %.2f\n", keliling);      return 0;  } |

Table 37 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | jarijari = input()  if len(jarijari.split()) == 2:      jarijari, tinggi = map(float, jarijari.split())  elif len(jarijari.split()) == 1:      jarijari = float(jarijari)      tinggi = float(input())  phi = 22.0 / 7.0  volume   = phi \* jarijari\*\*2 \* tinggi  luas     = 2 \* phi \* jarijari \* tinggi + 2 \* phi \* jarijari\*\*2  keliling = 2 \* phi \* jarijari  print(f"Volume   = {volume:.2f}")  print(f"Luas     = {luas:.2f}")  print(f"Keliling = {keliling:.2f}") |

Table 38 Source Code Bahasa Python Soal 4

## Output Program

Gambar 37 Screenshot Output Bahasa C Soal 4

3

Gambar 38 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf().*
* Pada baris 3: syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 5-6: Dua variabel *jarijari* dan tinggi dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. *Phi* memiliki nilai *22.0 / 7.0*
* Pada baris 8-9: Fungsi *scanf()* digunakan untuk mengambil input. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal.
* Pada baris 15-19: Syntax printf (“ “) adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Format *(%.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma. Fungsi ( *;* ) atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: *input()* menerima input sebagai satu string.
* Pada baris 3-13: : Dua variabel *jarijari* dan *tinggi* dideklarasikan sebagai tipe data (*float)* untuk menyimpan angka desimal. *Phi* memiliki nilai *22.0 / 7.0*. Input akan dipecah menggunakan *split()* dan dikonversi ke float dengan *map().*
* Pada baris 15-17: Syntax *(f’ ’)* digunakan untuk menyisipkan variabel langsung ke dalam string. Format *(.2f)* berarti hasil akan ditampilkan sebagai angka desimal dengan dua angka setelah koma.

# SOAL 5

1. Buatlah sebuah program yang digunakan untuk menghitung alas, tinggi keliling dan luas segitiga pythagoras, Jika yang diketahui hanya A dan B.

Sebuah gambar berisi garis, diagram

Deskripsi dibuat secara otomatis

|  |  |
| --- | --- |
| **Input** | **Output** |
| 40 41 | Alas = 9 cm  Tinggi = 40 cm  Keliling = 90 cm  Luas = 180 cm^2 |
| 16  65 | Alas = 63 cm  Tinggi = 16 cm  Keliling = 144 cm  Luas = 504 cm^2 |

## 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>  #include <math.h>  int main()  {      int A, B;        scanf("%d", &A);      scanf("%d", &B);      int alas = sqrt(B \* B - A \* A);      int keliling = A + B + alas;      float luas   = 0.5 \* A \* alas;      printf("Alas     = %d cm\n", alas);      printf("Tinggi   = %d cm\n", A);      printf("Keliling = %d cm\n", keliling);      printf("Luas     = %.f cm^2\n", luas);      return 0;  } |

Table 39 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | import math  sisiTinggi = input()  if len(sisiTinggi.split()) == 2:      sisiTinggi, sisiMiring = map(int, sisiTinggi.split())  elif len(sisiTinggi.split()) == 1:      sisiTinggi = int(sisiTinggi)      sisiMiring = int(input())  alas     = int(math.sqrt(sisiMiring\*\*2 - sisiTinggi\*\*2))  keliling = sisiTinggi + sisiMiring + alas  luas     = int(0.5 \* sisiTinggi \* alas)  print(f"Alas     = {alas} cm")  print(f"Tinggi   = {sisiTinggi} cm")  print(f"Keliling = {keliling} cm")  print(f"Luas     = {luas} cm^2") |

Table 40 Source Code Bahasa Python Soal 5

## Output Program

Gambar 39 Screenshot Output Bahasa C Soal 5

3

Gambar 40 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1-2: syntax *#include* digunakan untuk mengimpor library standart input output agar bisa menggunakan perintah seperti *printf().* *#include <math.h>* digunakan untuk fungsi matematika seperti *sqrt* yang menghitung akar kuadrat.
* Pada baris 8-14: Fungsi *scanf()* digunakan untuk mengambil input. Tipe data(*float)* untuk menyimpan angka desimal.
* Pada baris 16-21: Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Format spesifier *(%.f)* digunakan untuk menampilkan nilai sebagai bilangan bulat (tanpa desimal). Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: *Import math* library ini diimpor untuk menggunakan fungsi matematika, khususnya math.sqrt untuk menghitung akar kuadrat.
* Pada baris 3: *input()* menerima input sebagai satu string.
* Pada baris 5-13: Dua variabel *sisiTinggi dan sisiMiring*. Input akan dipecah menggunakan *split()* dan dikonversi dengan *map().*
* Pada baris 15-18: Syntax *(f’ ’)* digunakan untuk menyisipkan variabel langsung ke dalam string.

# MODUL 3 : Kondisional

# SOAL 1

1. Buatlah sebuah program untuk mengurutkan 3 angka inputan dari nilai terkecil ke terbesar menggunakan metode kondisional.

Input merupakan angka secara acak dengan output adalah hasil dari pengurutan.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 36 12 | 12 36 |
| 5 6 | 5 6 |
| 94 65 | 65 94 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  2  13  14  15  16  17  18  19  20  21  22 | #include <stdio.h>  int main()  {  int a,b,c;  scanf("%d %d %d",&a ,&b, &c);      if(a>b && a>c)  {      if(b>c){printf("%d %d %d",c, b, a);}      else {printf("%d %d %d",b, c, a);}  }      else if (b>c && b>a)  {      if(a>c){printf("%d %d %d",c, a, b);}      else{printf("%d %d %d", a, c, b);}  }        else if(c>a && c>b)  {      if(a>b){printf("%d %d %d",b, a, c);}      else {printf("%d %d %d",a, b, c);}  }      return 0;  } |

Table 41 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | a, b, c=map(int, input().split())  if a>=b>=c: print(f"{c} {b} {a}")  elif a>=c>=b : print(f"{b} {c} {a}")  elif b>=c>=a : print(f"{a} {c} {b}")  elif b>=a>=c : print(f"{c} {a} {b}")  elif c>=a>=b : print(f"{b} {a} {c}")  elif c>=b>=a : print(f"{a} {b} {c}") |

Table 42 Source Code Bahasa Python Soal 1

## Output Program

Gambar 41 Screenshot Output Bahasa C Soal 1

Gambar 42 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int a,b,c ()* menyatakan variabel *a,b,c*  bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: Fungsi *scanf("%d %d %d",&a ,&b, &c);* Membaca bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel *a,b,c*.
* Pada baris 6-23: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *a, b, c=map(int, input().split())* membaca tiga angka bilangan bulat dari input pengguna, dipisahkan oleh spasi, lalu menyimpannya ke dalam variabel *a, b, dan c.*
* Pada baris 3-12: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan

# SOAL 2

Sebuah gambar berisi teks, cuplikan layar, nomor, Font

Deskripsi dibuat secara otomatis2. Buatlah sebuah program untuk melakukan konversi nilai seperti pada tabel nilai berikut !

|  |  |
| --- | --- |
| **Input** | **Output** |
| 50 | D |
| 75 | B |
| 68 | C |
| 98 | A |
| 49 | E |

## 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 | #include <stdio.h>  int main (){      int nilai;      for (int i = 0; i < 5; i++)      {          scanf("%d", &nilai);          if(nilai >= 80)          {              printf("A\n");          }              else if(nilai >= 70 && nilai < 80)          {              printf("B\n");          }              else if (nilai >= 60 && nilai < 70)          {              printf("C\n");          }              else if(nilai >= 50 && nilai < 60)          {              printf("D\n");          }              else          {              printf("E\n");          }      }          return 0;  } |

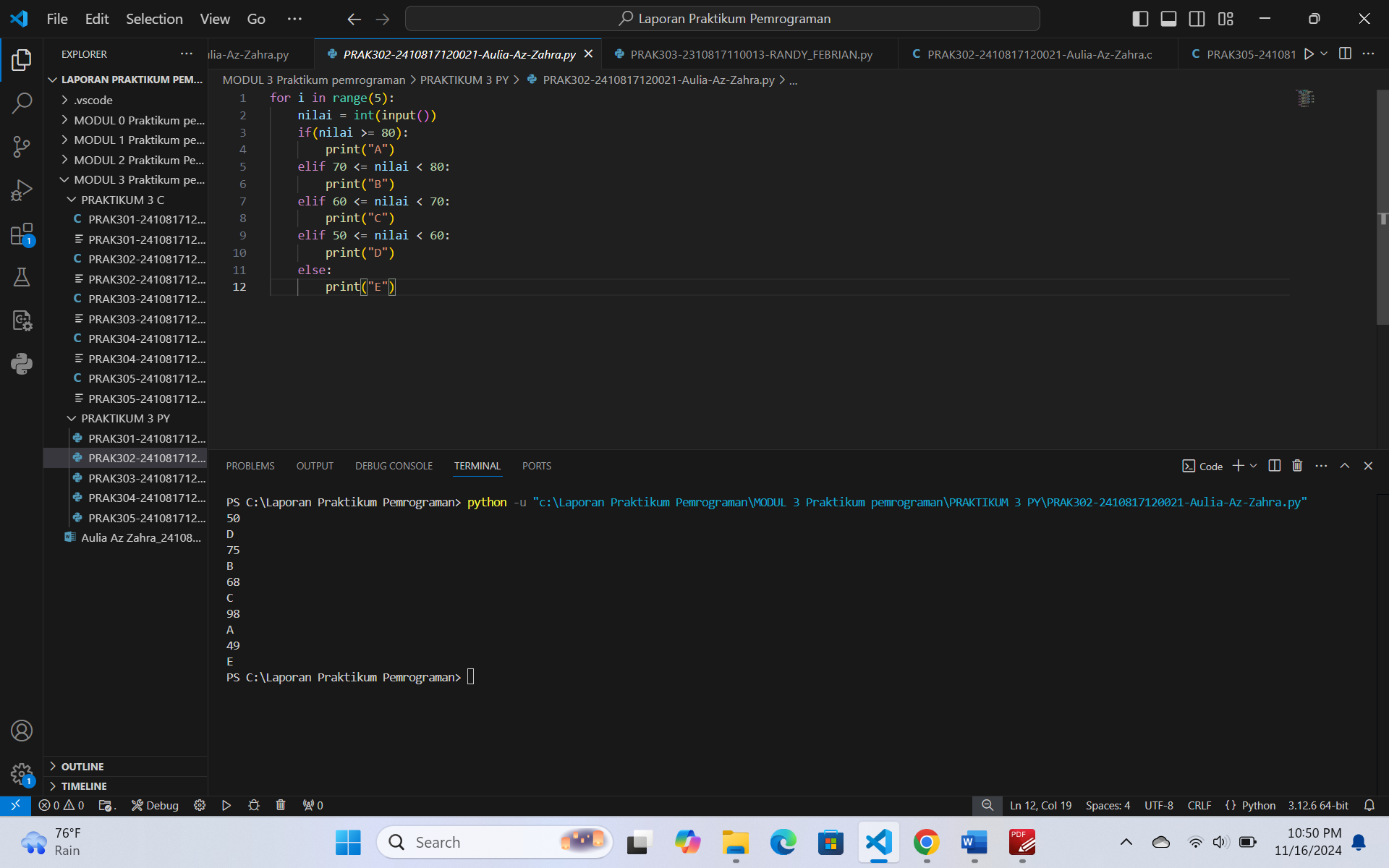
Table 43 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | i = 0  while(i < 3):      angka1, angka2 = input().split()      if(angka1 < angka2):          print(angka1, angka2)      else:          print(angka2, angka1)      i = i + 1 |

Table 44 Source Code Bahasa Python Soal 2

## Output Program

Gambar 43 Screenshot Output Bahasa C Soal 2

****

Gambar 44 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 3: *int nilai ()* menyatakan variabel *nilai* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 4: *for (int i = 0; i < 5; i++)* loop *for* digunakan untuk melakukan iterasi sebanyak 5 kali. Variabel *i* digunakan sebagai penghitung aritmatika.
* Pada baris 6: *scanf("%d", &nilai);* Membaca nilai bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel nilai.
* Pada baris 7-29: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *for i in range (5):* membuat loop *for* sebanyak 5 kali, menghasilkan angka dari 0 hingga 4 dengan total 5 iterasi.
* Pada baris 2: Fungsi *nilai = int(input())* menerima input dalam bentuk string lalu dikonversikan ke tipe data integer untuk operasi perbandingan angka.
* Pada baris 3-12: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan

# SOAL 3

3. Pak Soni meminta kepada Anda untuk membuat sebuah program sebagai berikut. Program akan menerima sebuah bilangan bulat N. Jika N adalah bilangan bulat positif, cetak positif. Jika N adalah bilangan bulat negatif, cetak negatif. Selain itu (yakni jika N adalah nol), cetak nol.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 50 | positif |
| -3000 | negatif |
| 0 | nol |

## 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 | #include <stdio.h>  int main()  {      int N;      for(int i = 0; i < 3; i++)      {          scanf("%d", &N);          if(N > 0)          {              printf("positif\n");          }              else if(N < 0)          {              printf("negatif\n");          }              else          {              printf("nol");          }      }          return 0;  } |

Table 45 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | for i in range(3):      nilai = int(input())      if(nilai > 0):          print("positif")      elif(nilai < 0):          print("negatif")      else:          print("nol") |

Table 46 Source Code Bahasa Python Soal 3

## Output Program

Gambar 45 Screenshot Output Bahasa C Soal 3

3

Gambar 46 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int N;* menyatakan variabel *N* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *for (int i = 0; i < 3; i++)* loop *for* digunakan untuk melakukan operasi aritmatika sebanyak 3 kali. Variabel *i* digunakan sebagai penghitung aritmatika.
* Pada baris 7: *scanf("%d", &n);* Membaca nilai bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel *N*.
* Pada baris 9-23: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *for i in range (3):* membuat loop *for* sebanyak 3 kali, menghasilkan angka dari 0 hingga 2 dengan total 3 iterasi.
* Pada baris 2: Fungsi *nilai = int(input())* menerima input dalam bentuk string lalu dikonversikan ke tipe data integer untuk operasi perbandingan angka.
* Pada baris 3-8: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 4

4. Pak Ranzi ingin meminta anda untuk membuatkan program untuk membaca ejaan dari bilangan cacah agar mempermudah pekerjaannya.

Format Masukan :

Sebuah baris berisi satu buah bilangan cacah dan batas maksimal 99. ( a >= 0 < 100 )

Format Keluaran :

Sebuah baris berisi sebuah bilangan cacah yang merupakan Ejaan dari bilangan tersebut atau batas limit bilangan.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 | Satuan |
| 0 | Nol |
| 100 | Anda Menginput Melebihi Limit Bilangan |
| 62 | Puluhan |
| 13 | Belasan |

## 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 | #include <stdio.h>  int main()  {      int a;      for(int i = 0; i < 5; i++)      {          scanf("%d", &a);          if (a < 0)          {              break;          }              else if (a == 0)          {              printf("Nol\n");          }              else if(a > 0 && a < 10)          {              printf("Satuan\n");          }              else if (a > 10 && a < 20)          {              printf("Belasan\n");          }              else if (a >= 20 && a < 100 || a == 10)          {              printf("Puluhan\n");          }              else          {              printf("Anda menginput Melebihi Limit Bilangan\n");          }      }      return 0;  } |

Table 47 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | for i in range(5):      nilai = int(input())      if(nilai < 0):          break      elif(nilai == 0):          print("Nol")      elif(0 < nilai < 10):          print("Satuan")      elif(10 < nilai < 20):          print("Belasan")      elif(20 <= nilai < 100 or nilai == 10):          print("Puluhan")      elif(nilai <= 100):          print("Anda Menginput Melebihi Limit Bilangan") |

Table 48 Source Code Bahasa Python Soal 4

## Output Program

Gambar 47 Screenshot Output Bahasa C Soal 4

3

Gambar 48 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int a;* menyatakan variabel *a* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *for (int i = 0; i < 5; i++)* loop *for* digunakan untuk melakukan operasi sebanyak 5 kali. Variabel *i* digunakan sebagai penghitung operasi aritmatika.
* Pada baris 7: *scanf("%d", &a);* Membaca bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel *a*.
* Pada baris 8-34: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *for i in range (5):* membuat loop *for* sebanyak 3 kali, menghasilkan angka dari 0 hingga 4 dengan total 5 operasi.
* Pada baris 2: Fungsi *nilai = int(input())* menerima input dalam bentuk string lalu dikonversikan ke tipe data integer untuk operasi perbandingan angka.
* Pada baris 3-8: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 5

5. Buat program untuk mengkonversikan jumlah detik ke dalam jam, menit, dan detik.

Format Masukan :

Sebuah bilangan yang merepresentasikan detik

Format Keluaran :

Sebuah baris berisi angka hasil konfersi jam, menit, dan detik. (dengan format

**jam:menit:detik**)

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3600 | 01:00:00 |
| 1432 | 00:23:52 |
| 8453 | 02:20:53 |
| 21542 | 05:59:02 |
| 125478 | 1 hari 10:51:18 |

## 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 | #include <stdio.h>  int main(){  int input\_detik;        for(int i = 0; i < 5; i++)      {          int hari = 0, menit = 0, detik = 0, jam = 0;          scanf("%d", &input\_detik);          while(input\_detik > 0)          {                  if(input\_detik >= 86400)              {                  hari++;                  input\_detik = input\_detik - 86400;              }                  else if(input\_detik >= 3600 && input\_detik <= 86400.)              {                  jam++;                  input\_detik = input\_detik - 3600;              }                  else if(input\_detik >= 60 && input\_detik <= 3600)              {                  menit++;                  input\_detik = input\_detik - 60;              }                  else              {                  detik++;                  input\_detik--;              }          }              if(hari >= 1)          {              printf("%d hari %.2d:%.2d:%.2d\n", hari, jam, menit, detik);          }              else          {              printf("%.2d:%.2d:%.2d\n", jam, menit, detik);          }      }      return 0;  } |

Table 49 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  22  23 | for i in range(5):      input\_detik = int(input())      hari, jam, menit, detik = 0, 0, 0, 0      while input\_detik > 0:          if input\_detik >= 86400:              hari = hari + 1              input\_detik = input\_detik - 86400          elif input\_detik >= 3600:              jam = jam + 1              input\_detik = input\_detik - 3600          elif input\_detik >= 60:              menit = menit + 1              input\_detik = input\_detik - 60          else:              detik = detik + 1              input\_detik = input\_detik - 1      if hari >= 1:          print("%d hari %.2d:%.2d:%.2d" % (hari, jam, menit, detik))      else:          print("%.2d:%.2d:%.2d" % (jam, menit, detik)) |

Table 50 Source Code Bahasa Python Soal 5

## Output Program

Gambar 49 Screenshot Output Bahasa C Soal 5

3

Gambar 50 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 3: Fungsi *int input\_detik;* untuk menyimpan input dalam jumlah detik.
* Pada baris 5: *for (int i = 0; i < 5; i++)* loop *for* digunakan untuk melakukan operasi sebanyak 5 kali. Variabel *i* digunakan sebagai penghitung operasi aritmatika.
* Pada baris 8: *scanf("%d", &input\_detik);* menerima input berupa detik.
* Pada baris 9: Loop *while* dijalankan selama nilai *input\_detik > 0* . Setiap iterasi akan mengurangi *input\_detik* sesuai dengan konversi waktu.
* Pada baris 11-42: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. *Printf(”%d hari %.2d:%.2d:%.2d\n”, hari, jam, menit, detik);* menampilkan hasil operasi dalam format waktu dengan jumlah hari. *Printf(”%.2d:%.2d:%.2d\n”, jam, menit, detik);* menampilkan hasil operasi dalam format waktu tanpa hari . Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *for i in range (5):* membuat loop *for* sebanyak 3 kali, menghasilkan angka dari 0 hingga 4 dengan total 5 iterasi.
* Pada baris 2: Fungsi *input\_detik = int(input())* menerima input dalam bentuk jumlah detik.
* Pada baris 4: Loop *while* dijalankan selama nilai *input\_detik > 0* . Setiap iterasi akan mengurangi *input\_detik* sesuai dengan konversi waktu.
* Pada baris 5-21: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. *Printf(”%d hari %.2d:%.2d:%.2d\n”, hari, jam, menit, detik);* menampilkan hasil operasi dalam format waktu dengan jumlah hari. *Printf(”%.2d:%.2d:%.2d\n”, jam, menit, detik);* menampilkan hasil operasi dalam format waktu tanpa hari.

# MODUL 4 : Loop

# SOAL 1

1. Pak Dengklek merupakan seorang guru TK Suka Berhitung. Hari ini Pak Dengklek mengajarkan murid-muridnya bilangan kelipatan dengan cara bermain. Setiap murid diminta untuk berjejer dan menyebutkan angka sesuai urutannya kecuali bilangan kelipatan yang harus disebut dengan suatu simbol. Misalnya simbol yang harus disebutkan adalah bintang (\*) pada kelipatan 3 maka urutan yang disebut pada tiap anak menjadi: 1 2 \* 4 5 \* dan seterusnya. Buatlah program untuk membantu Pak Dengklek agar permainan dapat berjalan dengan baik jika jumlah anak didiknya adalah 50 anak.

Info:

Input pertama merupakan bilangan kelipatan yang dirubah menjadi simbol Input kedua merupakan simbol yang akan menggantikan bilangan tersebut Output merupakan bilangan 1-50 dengan bilangan kelipatan dirubah menjadi simbol.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6 \* | 1 2 3 4 5 \* 7 8 9 10 11 \* 13 14 15 16 17 \* 19 20 21 22 23 \* 25 26 27 28 29 \* 31 32 33 34 35 \* 37 38 39 40 41 \* 43 44 45 46 47 \* 49 50 |
| 3 # | 1 2 # 4 5 # 7 8 # 10 11 # 13 14 # 16 17 # 19 20 # 22 23 # 25 26 # 28 29 # 31 32 # 34 35 # 37 38 # 40 41 # 43 44 # 46 47 # 49 50 |
| 11 & | 1 2 3 4 5 6 7 8 9 10 & 12 13 14 15 16 17 18 19 20 21 & 23 24 25 26 27 28 29 30 31 32 & 34 35 36 37 38 39 40 41 42 43 & 45 46 47 48 49 50 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  2  13  14  15  16  17  18 | #include <stdio.h>  int main()  {      int n;      char s[50];      scanf("%d %s", &n, &s);      printf("\n");      for(int i = 1; i <= 50; ++i)      {          if (i % n == 0)          {              printf("%s ", s);              continue;          }          printf("%d ", i);      }      return 0;  } |

Table 51 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | n, s =input().split()  n = int(n)  print()  for x in range(1, 51):      if x % n == 0:          print(s, end=" ")          continue      print(x, end=" ") |

Table 52 Source Code Bahasa Python Soal 1

## Output Program

Gambar 51 Screenshot Output Bahasa C Soal 1

Gambar 52 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int n* menyatakan variabel *n* bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: Fungsi *char s [50]* sebagai array untuk menyimpan data.
* Pada baris 6: Fungsi *scanf("%d %s",&n ,&s);* Membaca bilangan bulat dari input pengguna.
* Pada baris 7: Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan.
* Pada baris 8: Fungsi  *for(int i = 1; i <= 50; ++i)* Memulai perulangan for dari 1 hingga 50.
* Pada baris 9-18: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *continue* melewati sisa instruksi dalam loop ini dan melanjutkan ke loop berikutnya. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *n, s =input().split()*membaca dua angka bilangan bulat dari input pengguna.
* Pada baris 2: Fungsi *n = int(n)* mengonversi n menjadi tipe data untuk bilangan bulat.
* Pada baris 4: Fungsi *for x in range(1, 51):* memulai loop *for* dengan *x* mulai dari 1 hingga 50.
* Pada baris 5-8: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *continue* melewati sisa instruksi dalam loop ini dan melanjutkan ke loop berikutnya. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan

# SOAL 2

2. Hari ini Pak Dengklek tidak bisa mengajar anak didiknya yang ada di TK Suka Berhitung karena sedang sakit sehingga harus digantikan oleh Bu Dengklek. Pak Dengklek berpesan kepada Bu Dengklek agar mengajarkan materi bilangan genap dan ganjil kepada muridnya. Bu Dengklek mempunyai metode agar materi bilangan ganjil genap dapat disampaikan dengan cara yang menyenangkan. Metodenya adalah dengan cara mengurutkan bilangan ganjil dari 1 sampai batas tertentu dan meyebutkan bilangan genapnya secara terbalik. Buatlah program untuk mempermudah Bu Dengklek menghitung bilangan tersebut!

Info:

Input merupakan batas maksimal dari bilangan yang akan dihitung Output baris pertama merupakan urutan bilangan ganjil dari 1 sampai batas maksimal Output baris kedua merupakan urutan bilangan genap dari batas maksimal sampai 2

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10 | 1 3 5 7 9  10 8 6 4 2 |
| 25 | 1 3 5 7 9 11 13 15 17 19 21 23 25  24 22 20 18 16 14 12 10 8 6 4 2 |
| 6 | 1 3 5  6 4 2 |

## 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 | #include <stdio.h>  int main()  {      int n;      scanf("%d", &n);      printf("\n");      for (int i = 1; i <= n;++i)      {          if(i % 2 == 1)          {              printf("%d ", i);          }      }      printf("\n");      for (int j = n; j >= 1; --j)      {          if(j % 2 == 0)          {              printf("%d ", j);          }      }      return 0;  } |

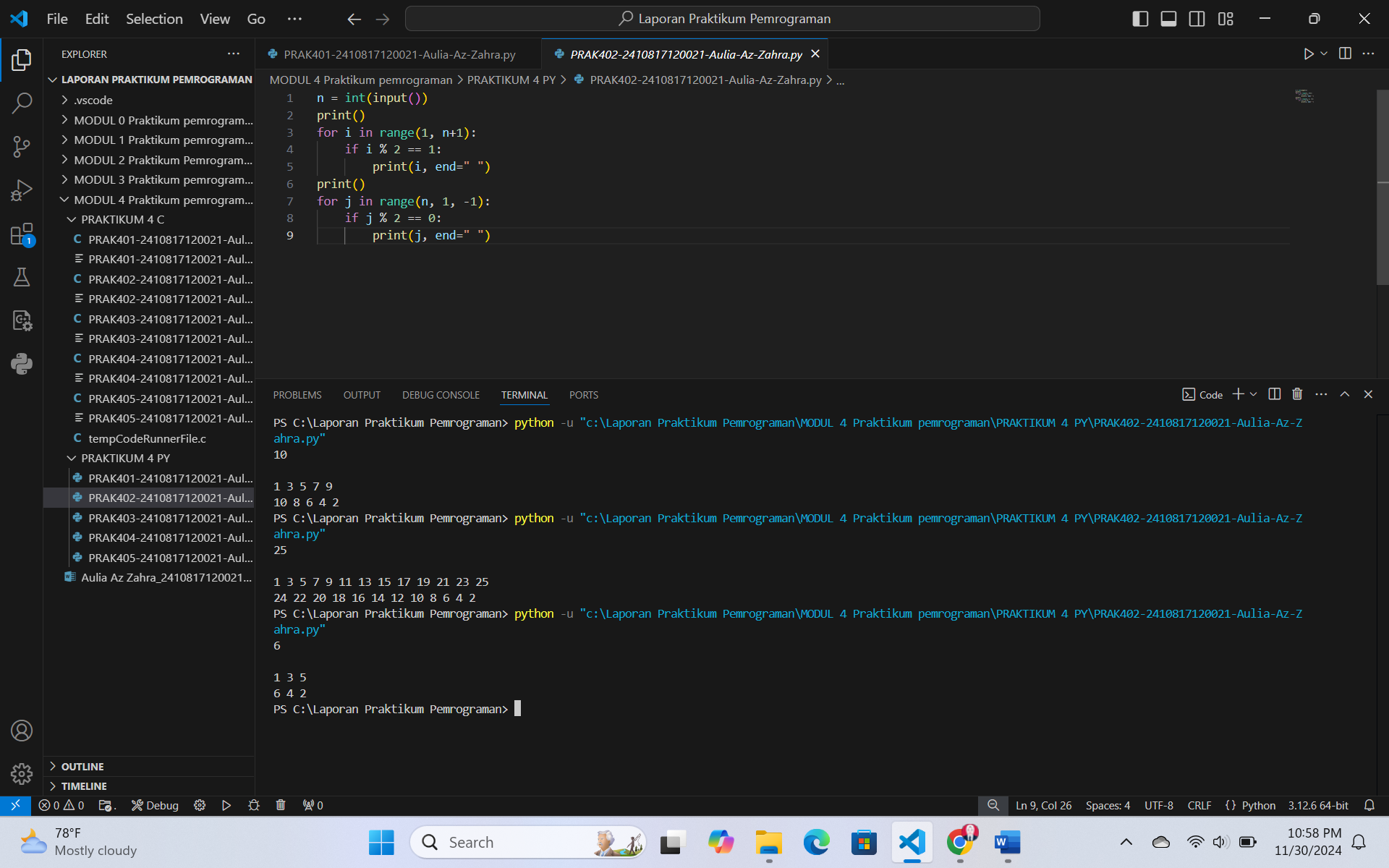
Table 53 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | n = int(input())  print()  for i in range(1, n+1):      if i % 2 == 1:          print(i, end=" ")  print()  for j in range(n, 1, -1):      if j % 2 == 0:          print(j, end=" ") |

Table 54 Source Code Bahasa Python Soal 2

## Output Program

Gambar 53 Screenshot Output Bahasa C Soal 2

****

Gambar 54 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int n* menyatakan variabel *n* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *scanf(”&d, &n”);* membaca nilai yang akan dimamasukkan sebagai bilangan bulat lalu menyimpannya ke dalam variabel *n*.
* Pada baris 4: *for (int i = 1; i <= n;++i)* melakukan loop dari *i = 1* hingga *1 = n* dengan menambah i sebanyak 1 setiap pengoperasian.
* Pada baris 8-23: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Fungsi *if-else / if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *n = int(input())* membaca input sebagai bilangan bulat lalu menyimpannya dalam variabel *n.*
* Pada baris 3: Fungsi *for i in range(1, n+1):* melakukan loop dari *i = 1* hingga *1 = i* .
* Pada baris 3-12: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 3

3. Pak Samson adalah seorang pengusaha sekaligus programmer ternama di Kota Xam. Pak Samson kesulitan membagi waktu antara membuat program dan mengurus usahanya, lalu Pak Samson menyuruh anda membuat program yang diminta oleh clientnya, permintaannya sebagai berikut : Buatlah program angka yang bersilangan, dan input terdiri dari angka pertama dan angka kedua, setiap angka yang bersilangan dibatasi dengan simbol . jika angka pertama lebih besar dari angka kedua maka program akan menampilkan dari terkecil ke terbesar dari batasan angka kedua. jika angka kedua lebih besar dari angka pertama maka program akan menampilkan dari terbesar ke terkecil dari batasan angka kedua. Untuk lebih jelasnya lihat tabel input output berikut :

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 7 | 3 7 - 4 6 - 5 5 - 6 4 - 7 3 |
| 7 3 | 7 3 - 6 4 - 5 5 - 4 6 - 3 7 |
| 95 100 | 95 100 - 96 99 - 97 98 - 98 97 - 99 96 - 100 95 |
| 23 17 | 23 17 - 22 18 - 21 19 - 20 20 - 19 21 - 18 22 - 17 23 |

## 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 | #include <stdio.h>  int main()  {      int n, m;      scanf("%d %d", &n, &m);      int i = n;      int j = m;      printf("\n");      while (i < m && j > n)      {          printf("%d %d - ", i, j);          i++;          j--;          if (i == m && j == n)          {              printf("%d %d", i, j);          }      }      while (i > m && j < n)      {          printf("%d %d - ", i, j);          i--;          j++;          if (i == m && j == n)          {              printf("%d %d", i, j);          }      }      return 0;  } |

Table 55 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | n, m =map(int, input().split())  i = n  j = m  print()  while i < m and j > n:      print(f"{i} {j} -", end=" ")      i += 1      j -=1      if i == m and j == n:          print(f"{i} {j}", end=" ")  while i > m and j < n:      print(f"{i} {j} -", end=" ")      i -= 1      j += 1      if i == m and j == n:          print(f"{i} {j}", end=" ") |

Table 56 Source Code Bahasa Python Soal 3

## Output Program

Gambar 55 Screenshot Output Bahasa C Soal 3

3

Gambar 56 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4:: *int n, m;* menyatakan variabel *n* dan *m* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: Fungsi *scanf("%d %d", &n ,&m);* Membaca bilangan bulat dari input pengguna lalu menyimpannya dalam variabel *n* dan *m*.
* Pada baris 6-7: *int i = n* dan *int j =* menyatakan variabel *i* dengan nilai *n* dan *j* dengan nilai *m.*
* Pada baris 8-30: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Fungsi *if-else / if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *n, m =map(int, input().split())*membaca bilangan bulat dari input lalu memisahkannya dengan spasi, dan menyimpannya dalam variabel *n* dan *m*.
* Pada baris 2-3: *i = n* dan *j = m* menyatakan variabel *i* dengan nilai *n* dan *j* dengan nilai *m.*
* Pada baris 3-8: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 4

4. Pa Jack mencari seorang programmer yang bisa membuatkan program kalkulator sederhana untuk dirinya, karena dia kesulitan menghitung hasil penjualannya. Buatlah sebuah kalkulator sederhana yang di minta sesuai dengan yang Pa Jack inginkan dengan output sebagai berikut :

Pilih program

1. Penjumlahan

2. Pengurangan

3. Perkalian

4. Pembagian

5. Exit

Masukkan Pilihan :

Masukkan nilai pertama :

Masukkan nilai kedua :

Hasil Pilihan antara Nilai Pertama dengan Nilai Kedua adalah Hasil

* Selama program belum memasukkan pilihan angka 5, maka program akan terus berjalan.
* Jika memasukkan angka 5 maka selanjutnya program selesai dan tampilkan : Terimakasih, telah menggunakan kalkulator NAMAANDA
* Jika memasukkan angka selain dari angka yang tertera mulai ulang programnya dan tampilkan : Input anda salah, silahkan coba lagi

|  |  |
| --- | --- |
| **Input** | **Output** |
| Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit Masukkan Pilihan : 3  Masukkan nilai pertama : 12  Masukkan nilai kedua : 5 | Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit Masukkan Pilihan : 3  Masukkan nilai pertama :12  Masukkan nilai kedua :5  Hasil perkalian antara 12.00 dengan 5.00  adalah 60.00 |
| Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit Masukkan Pilihan : 13 | Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit  Masukkan Pilihan : 13 Input anda salah,  silahkan coba lagi |
| Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit Masukkan Pilihan : 5 | Pilih program  1. Penjumlahan  2. Pengurangan  3. Perkalian  4. Pembagian  5. Exit  Masukkan Pilihan : 5 Terimakasih, telah menggunakan kalkulator NAMAANDA |

## 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 | #include <stdio.h>  int main()  {      int n;      float a, b;      while(1)      {          printf("\nPilih program\n1. penjumlahan\n2. Pengurangan\n3. Perkalian\n4. Pembagian\n5. Exit");          printf("\nMasukkan Pilihan : ");          scanf("%d", &n);          if(n < 1 || n > 5)          {printf("Input anda salah, silahkan coba lagi"); continue;}          else if (n == 5)          {printf("Terimakasih, telah menggunakan kalkulator AULIA AZ ZAHRA"); break; }            printf("Masukkan nilai pertama : ");          scanf("%lf", &a);          printf("Masukkan nilai kedua : ");          scanf("%lf", &b);          if (n == 1)          {              printf("Hasil penjumlahan antara %.2lf dengan %.2lf adalah %.2lf\n\n", a, b, a+b);          }          else if (n == 2)          {              printf("Hasil pengurangan antara %.2lf dengan %.2lf adalah %.2lf\n\n", a, b, a-b);          }          else if (n == 3)          {              printf("Hasil perkalian antara %.2lf dengan %.2lf adalah %.2lf\n\n", a, b, a\*b);          }          else if (n == 4)          {              printf("Hasil pembagian antara %.2lf dengan %.2lf adalah %.2lf\n\n", a, b, a/b);          }      }      return 0;  } |

Table 57 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 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 | while True:      print("\nPilih program\n1. Penjumlahan\n2. Pengurangan\n3. Perkalian\n4. Pembagian\n5. Exit")      n = int(input("Masukkan Pilihan : "))      if n < 1 or n > 5:          print("Input anda salah, silahkan coba lagi")          continue      elif n == 5:          print("Terimakasih, telah menggunakan kalkulator AULIA AZ ZAHRA")          break        a = float(input("Masukkan nilai pertama : "))      b = float(input("Masukkan nilai kedua : "))      if n == 1:          Hasil = a + b          print(f"Hasil penjumlahan antara {a:.2f} dengan {b:.2f} adalah {Hasil:.2f}")      elif n == 2:          Hasil = a - b          print(f"Hasil pengurangan antara {a:.2f} dengan {b:.2f} adalah {Hasil:.2f}")      elif n == 3:          Hasil = a \* b          print(f"Hasil perkalian antara {a:.2f} dengan {b:.2f} adalah {Hasil:.2f}")      elif n == 4:          Hasil = a / b          print(f"Hasil pembagian antara {a:.2f} dengan {b:.2f} adalah {Hasil:.2f}") |

Table 58 Source Code Bahasa Python Soal 4

## Output Program

Gambar 57 Screenshot Output Bahasa C Soal 4

3

Gambar 58 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int n;* menyatakan variabel sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *float a, b;* menyatakan tipe data berupa float.
* Pada baris 6: *while(1)*  memulai infinite loop untuk menjalankan program akan terus berjalan sampai ada perintah *break.*
* Pada baris 8-40: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. Fungsi *continue* melewati sisa instruksi dalam loop ini dan melanjutkan ke loop berikutnya. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *while true:* memulai infinite loop untuk menjalankan program terus-menerus sampai ada perintah *break.*
* Pada baris 2: Fungsi *n = int(input("Masukkan Pilihan : "))* memasukkan angka pilihan lalu menyimpannya dalam veriabel *n* .
* Pada baris 12-13: Variabel *a, b* dinyatakan sebagai tipe data float.
* Pada baris 5-26: Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Meskipun variabel bertipe float, penggunaan *(%.f)* akan menghasilkan output tanpa bagian desimal. Fungsi *continue* melewati sisa instruksi dalam loop ini dan melanjutkan ke loop berikutnya. Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.

# SOAL 5

5. Buatlah sebuah program untuk menghitung jumlah kelipatan pada tiap baris seperti contoh

kasus.

Info:

Input baris pertama, banyaknya n.

Input baris kedua, kelipatan.

Output adalah hasil perhitungan dari masing-masing kelipatan pada tiap baris dan output

baris terakhir merupakan jumlah pada masing-masih baris.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 2 | (1 \* 2) = 2  (2 \* 2) + (1 \* 2) = 6  (3 \* 2) + (2 \* 2) + (1 \* 2) = 12 20 |
| 5 3 | (1 \* 3) = 3  (2 \* 3) + (1 \* 3) = 9  (3 \* 3) + (2 \* 3) + (1 \* 3) = 18  (4 \* 3) + (3 \* 3) + (2 \* 3) + (1 \* 3) = 30  (5 \* 3) + (4 \* 3) + (3 \* 3) + (2 \* 3) + (1 \* 3) =  45105 |
| 2 3 | (1 \* 3) = 3  (2 \* 3) + (1 \* 3) = 9 12 |

## 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 m, n;      scanf("%d %d", &m, &n);      int all\_result = 0;      for (int i = 1;i <= m; ++i)      {          int result = 0;          printf("\n");          for (int j = i; j >= 1; --j)          {              int term = j \* n;              result += term;              all\_result += term;              printf("(%d \* %d)", j, n);              if (j > 1)              {                  printf(" + ");              }          }          printf(" = %d", result);      }          printf("\n%d", all\_result);        return 0;  } |

Table 59 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | m, n = map(int,input().split())  all\_result = 0  print()  for i in range(1,m + 1):      result = 0      for j in range(i, 0, -1):          term = j \* n          result += term          all\_result += term          print(f"({j} \* {n})", end=" ")          if j > 1:              print("+", end=" ")      print(f"= {result}", end=" ")      print()  print(all\_result) |

Table 60 Source Code Bahasa Python Soal 5

## Output Program

Gambar 59 Screenshot Output Bahasa C Soal 5

3

Gambar 60 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int m, n;* menyatakan variabel *m, n*  sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *scanf(”&d &d, m&, &n”);* membaca nilai yang akan dimasukkan sebagai bilangan bulat lalu menyimpannya ke dalam variabel m, *n*.
* Pada baris 6: *int all\_result = 0;* variabel dinyatakan sebagai bilangan bulat untuk menyimpan total keseluruhan.
* Pada baris 7: *for (int j = i; j >= 1; --j)* memulai loop untuk operasi dari *i = 1* hingga *i = m.*
* Pada baris 9: Fungsi *int result = 0;* untuk menyimpan penjumlahan pada baris tertentu.
* Pada baris 10-23: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Fungsi *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *if-else / if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *m, n =map(int, input().split())*membaca bilangan bulat dari input lalu memisahkannya dengan spasi, dan menyimpannya dalam variabel *m* dan *n*.
* Pada baris 2: *int all\_result = 0;* variabel dinyatakan sebagai bilangan bulat untuk menyimpan total keseluruhan*.*
* Pada baris 5: *for i range(1,m = 1):* memulai loop untuk operasi dari *i = 1* hingga *i = m.*
* Pada baris 6: Fungsi *result = 0* untuk menyimpan penjumlahan pada baris tertentu.
* Pada baris 6: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# MODUL 5 : Fungsi

# SOAL 1

1. Pak Roza merupakan seorang guru SD Selalu Ngoding. Hari ini Pak Roza mengajarkan murid-muridnya tentang angka maksimal dengan cara memperbaiki code yang kurang. Agar lebih efektif Pak Roza menyediakan code nya terlebih dahulu. Jadi Lengkapilah code di bawah ini dan buat dalam bahasa Python nya !

Info:

|  |
| --- |
| #include <stdio.h>  //Buatlah Function Disini  int main() {  int a, b, c, d;  scanf("%d %d %d %d", &a, &b, &c, &d);  int hasil = MaxBilangan(a, b, c, d); printf("%d", hasil);  return 0;  } |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1 3 4 2 | 4 |
| 7 5 3 9 | 9 |
| 11 23 51 49 | 51 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  2  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | #include <stdio.h>  //Buatlah Function Disini  int MaxBilangan(int a, int b, int c, int d)  {      int Max;      if (a >= b && a >= c && a >= d)      {          Max = a;      }      else if(b >= a && b >= c && b >= d)      {          Max = b;      }      else if(c >= a && c >= a && c >= d)      {          Max = c;      }      else if(d >= a && d >= b && d >= c)      {          Max = d;      }      return Max;  }  int main()  {      int a, b, c, d;      scanf("%d %d %d %d", &a, &b, &c, &d);      int hasil = MaxBilangan(a, b, c, d);      printf("%d", hasil);      return 0;  } |

Table 61 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | #Buatlah Function Disini  def MaxBilangan(a, b, c, d):      if(a >= b and a >= c and a >= d):          return a      elif(b >= a and b >= c and b >= d):          return b      elif (c >= a and c >= b and c>= d):          return c      elif (d >= a and d >= b and d >= c):          return d    a, b, c, d = map(int, input().split())  hasil = MaxBilangan(a, b, c, d)  print(hasil) |

Table 62 Source Code Bahasa Python Soal 1

## Output Program

Gambar 61 Screenshot Output Bahasa C Soal 1

Gambar 62 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 3: *int MaxBilangan(int a, int b, int c, int d)* untuk menerima input.
* Pada baris 6-23: Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.
* Pada baris 25: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 27: *int a,b,c,d ()* menyatakan variabel *a,b,c,d*  bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 28: Fungsi *scanf("%d %d %d %d",&a ,&b, &c, &d);* Membaca bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel *a,b,c,d*.
* Pada baris 30-32: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *def MaxBilangan(a, b, c, d):* menyatakan tiga variabel *a, b, c, d.*
* Pada baris 3-10: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.
* Pada baris 10: Fungsi *a, b, c, d = map(int, input().split())* membaca tiga angka bilangan bulat dari input pengguna, dipisahkan oleh spasi, lalu menyimpannya ke dalam variabel *a, b, c, d.*
* Pada baris 14: Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 2

2. Jarak Pulau Samosir adalah jarak dari suatu titik menuju suatu titik lainnya pada suatu sistem koordinat Kartesius dengan menyusuri bagian vertikal dan horizontal, tanpa pernah kembali lagi. Secara sederhana, sama dengan jumlah dari selisih absis dan selisih ordinat. Dengan kata lain, jarak Pulau Samosir = |x1 - x2| + |y1 - y2|.

Pak Ranzi ingin pergi dari koordinat (x1, y1) menuju (x2, y2). Tentukan jarak Pulau Samosir yang harus ditempuh Pak Ranzi.

Format Masukan :

Sebuah baris berisi empat buah bilangan bulat x1, y1, x2, dan y2.

Format Keluaran :

Sebuah baris berisi sebuah bilangan bulat yang merupakan jarak Pulau Samosir dari kedua titik tersebut.

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int hitung(int nilai1, int nilai2){  //Lengkapi Function ini  } int mutlak(int angka){  //Lengkapi Function ini }  int main() {  int a,b,c,d;  scanf("%d",&a);  scanf("%d",&c);  scanf("%d",&b);  scanf("%d",&d);  Hasil = hitung(a,b) + hitung(c,d);  printf("%d",mutlak(Hasil));  return 0;  } |

|  |  |
| --- | --- |
| **Input** | **Output** |
| -1 -1 1 1 | 4 |
| -5 6 -4 2 | 5 |
| 1 2 3 4 | 4 |

## 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 | #include <stdio.h>  #include <stdlib.h>  int hitung(int nilai1, int nilai2)  {      //Lengkapi Function ini      return abs(nilai1 - nilai2);  }  int mutlak(int angka)  {      //Lengkapi Function ini      return abs(angka);  }  int main()  {      int a,b,c,d;      scanf("%d", &a);      scanf("%d", &c);      scanf("%d", &b);      scanf("%d", &d);      int Hasil = hitung(a,b)+hitung(c,d);      printf("%d", mutlak(Hasil));      return 0;    } |

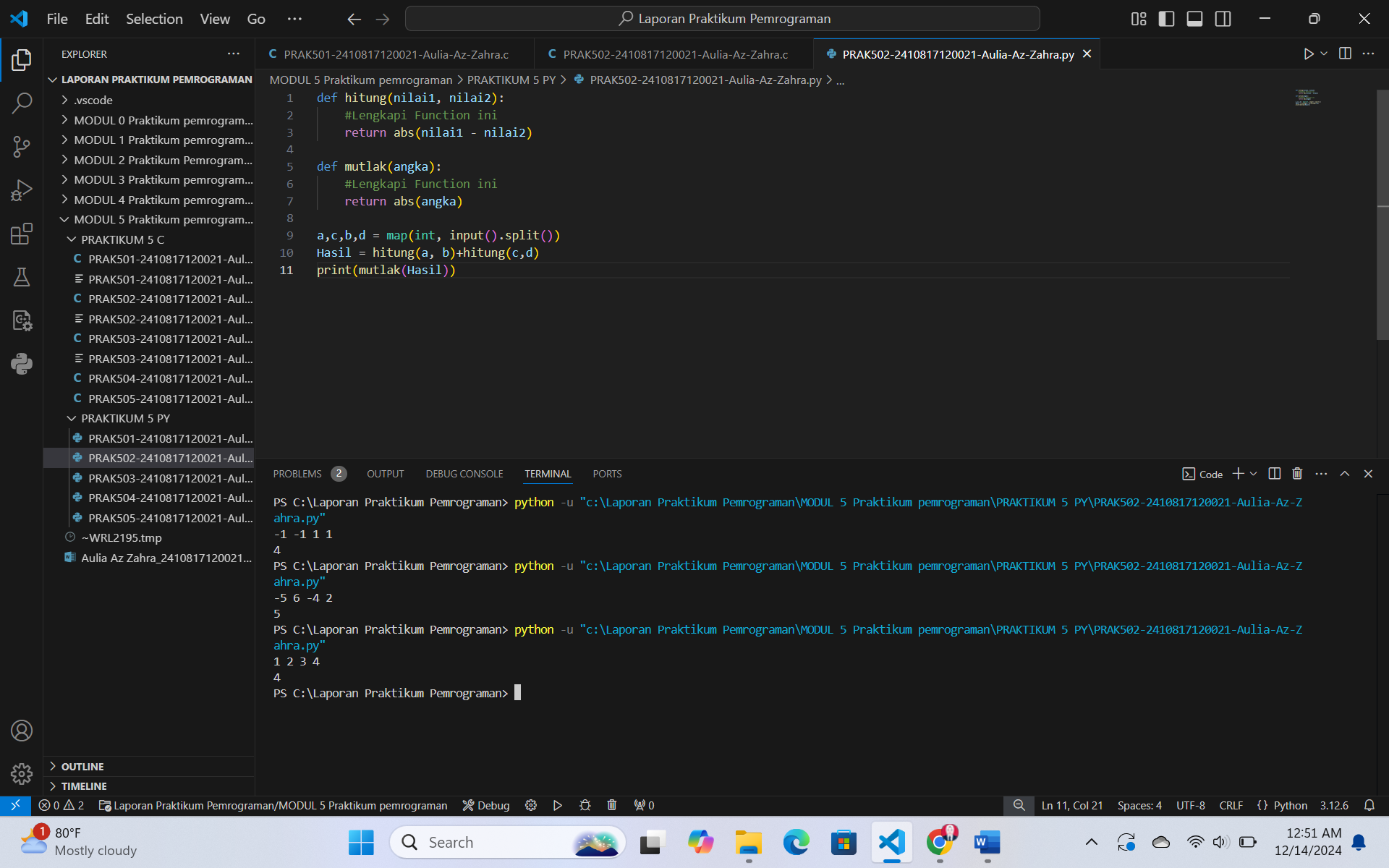
Table 63 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | def hitung(nilai1, nilai2):      #Lengkapi Function ini      return abs(nilai1 - nilai2)  def mutlak(angka):      #Lengkapi Function ini      return abs(angka)  a,c,b,d = map(int, input().split())  Hasil = hitung(a, b)+hitung(c,d)  print(mutlak(Hasil)) |

Table 64 Source Code Bahasa Python Soal 2

## Output Program

Gambar 63 Screenshot Output Bahasa C Soal 2

****

Gambar 64 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *#include <stdlib.h>* digunakan untuk mengimpor input output agar bisa menggunakan perintah seperti *abs()* untuk menghitung nilai mutlak.
* Pada baris 3: *int hitung(int nilai1, int nilai2)* untuk menerima input.
* Pada baris 13: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 15: *int a,b,c,d;* menyatakan variabel *a,b,c,d*  bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 16-26: Fungsi *scanf()* digunakan untuk mengambil input. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *def hitung(nilai1, nilai2)* menyatakan dua variabel *nilai1, nilai2.*
* Pada baris 9: Fungsi *a, b, c, d = map(int, input().split())* membaca tiga angka bilangan bulat dari input pengguna, dipisahkan oleh spasi, lalu menyimpannya ke dalam variabel *a, b, c, d.*
* Pada baris 11 : Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 3

3. Pak Roni Seorang master matematika yang sangat handal, Pak Roni menyuruh anda untuk membuatkan program menentukan nilai terbesar dan terkecil. Pak Roni memiliki Angka N buah bilangan bulat. Di antara bilangan-bilangan tersebut, tentukan bilangan terbesar dan bilangan terkecil.

**Format Masukan**

Baris pertama/awal berisi sebuah bilangan bulat N. Baris setelahnya berisi N buah bilangan bulat.

**Format Keluaran**

Sebuah baris berisi X Y, dengan X adalah bilangan terbesar dan Y adalah bilangan terkecil.

|  |
| --- |
| #include <stdio.h>  int maksimal(int a, int b){  // Lengkapi Function ini }  int minimal(int a, int b){  // Lengkapi Function ini  }  int main(){  int batas = 0;  int maks = -100000;  int minim = 100000;  int bilangan; scanf("%d", &bilangan);  while(batas < bilangan){ int nilai;  scanf("%d", &nilai);  maks = maksimal(maks, nilai);  minim = minimal(minim, nilai);  batas++;  }  printf("%d %d",maks,minim);  } |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  12 34 -5 -3 19 | 34 -5 |
| 8  1 -1 1 10 10 6 8 4 | 10 -1 |
| 10  1 -2 5 7 19 12 32 10 | 32 19 |

## 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 | #include <stdio.h>  int maksimal(int a, int b)  {      //Lengkapi Function ini      if (a > b){          return a;      }      else{          return b;      }  }  int minimal(int a, int b)  {      //Lengkapi Function ini      if (a < b)      {          return a;      }      else      {          return b;      }  }  int main()  {      int batas = 0;      int maks = -100000;      int minim = 100000;      int bilangan;      scanf("%d", &bilangan);      while(batas < bilangan)      {          int nilai;          scanf("%d", &nilai);          maks = maksimal(maks, nilai);          minim = minimal(minim, nilai);          batas++;      }      printf("%d %d",maks,minim);      return 0;  } |

Table 65 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | def maksimal(a, b):      #Lengkapi Function ini      if a > b:          return a      else:          return b  def minimal(a,b):      #Lengkapi Function ini      if(a < b):          return a      else:          return b  batas = 0  maks = -100000  minim = 100000  bilangan = int(input())  nilai = list(map(int, input().split()))  while(batas < bilangan):      maks = maksimal(maks, nilai[batas])      minim = minimal(minim, nilai[batas])      batas += 1  print(f"{maks} {minim}") |

Table 66 Source Code Bahasa Python Soal 3

## Output Program

Gambar 65 Screenshot Output Bahasa C Soal 3

3

Gambar 66 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 3: Fungsi *int maksimal(int a, int b)* untuk menerima input.
* Pada baris [5-22][31]: Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 30-41: Fungsi *scanf()* digunakan untuk mengambil input. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *def maksimal(a, b):*  menyatakan dua variabel *a, b.*
* Pada baris 8: Fungsi *def minimal(a, b):*  menyatakan dua variabel *a, b.*
* Pada baris 19: Fungsi *nilai = list(map(int, input().split())* membaca angka bilangan bulat dari input pengguna, dipisahkan oleh spasi, lalu menyimpannya ke dalam variabel.
* Pada baris 3-24: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 4

4. Pa Jack ingin menguji kecerdasan Anda. Anda akan diminta untuk membalik representasi desimal dari beberapa bilangan bulat positif, dengan mengabaikan leading zero. Sebagai contoh, 45670 jika dibalik menjadi 07654; lalu karena leading zero diabaikan, maka akan dianggap menjadi 7654.

Pak Jack akan memberikan Anda dua buah bilangan bulat positif A dan B. Anda diminta untuk membalik representasi desimal kedua bilangan tersebut. Sebut saja hasil pembalikan representasi desimal keduanya sebagai A' dan B'. Kemudian, Anda diminta untuk menjumlahkan A' dan B'. Sebut saja hasil penjumlahannya sebagai C. Terakhir, Anda diminta untuk mencetak hasil pembalikan representasi desimal dari C.

Sebagai contoh, A adalah 1290 dan B adalah 452. Maka, A' dan B' secara berurut adalah 921 dan 254. Hasil penjumlahan A' dan B' adalah C, yaitu 921 + 254 = 1175. Bilangan yang dicetak adalah pembalikan dari C, yaitu 5711.

|  |
| --- |
| #include <stdio.h>  int reverse(){  // Lengkapi Function ini  }  int main() {  int A, B;  scanf("%d %d",&A,&B);  A=reverse(A);  B=reverse(B);  int C = A+B;  printf("%d",reverse(C));  } |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1290 452 | 5711 |
| 5430 1120 | 655 |
| 932 114 | 56 |

## 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 | #include <stdio.h>  int reverse(int n)  {      //Lengkapi Function ini      int reversed = 0;      while (n != 0)      {          int remainder = n % 10;          reversed = reversed \* 10 + remainder;          n /= 10;      }      return reversed;  }  int main()  {      int A, B;      scanf("%d %d", &A, &B);      A = reverse(A);      B = reverse(B);      int C = A + B;      printf("%d", reverse(C));      return 0;  } |

Table 67 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | def reverse(n):      #Lengkapi Function ini      reversed = 0      while n != 0:          remainder = n % 10          reversed = reversed \* 10 + remainder          n //= 10      return reversed  A, B = map(int, input().split())  A = reverse(A)  B = reverse(B)  C = A + B  print(reverse(C)) |

Table 68 Source Code Bahasa Python Soal 4

## Output Program

Gambar 67 Screenshot Output Bahasa C Soal 4

3

Gambar 68 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Fungsi *int reverse(int n)* menerima input n sebagai bilangan bulat dan berupa angka yang dibalik.
* Pada baris 5: Fungsi *int reversed = 0;* menyatakan variabel tersebut dengan nilai 0. Variabel ini digunakan untuk menyimpan angka yang dibalik.
* Pada baris 6-12: Fungsi *if-else if* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.
* Pada baris 14: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 17: *scanf("%d %d", &A, &B);* Membaca nilai bilangan bulat dari input pengguna, lalu menyimpannya dalam variabel tersebut.
* Pada baris 18-23: Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *def reverse(n)* menerima input n sebagai bilangan bulat dan berupa angka yang dibalik.
* Pada baris 3: Fungsi *reversed = 0;* menyatakan variabel tersebut dengan nilai 0. Variabel ini digunakan untuk menyimpan angka yang dibalik.
* Pada baris 4-8: Fungsi *if / else if / elif / else* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai.
* Pada baris 10: Fungsi *A, B =map(int, input().split())* membaca tiga angka bilangan bulat dari input pengguna, dipisahkan oleh spasi, lalu menyimpannya ke dalam variabel *A, B.*
* Pada baris 14: Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 5

5. Pak Denni meminta anda untuk melengkapi function berikut supaya programnya bisa dijalankan dengan baik dan benar.

Format Masukkan : yang pertama tahun lahir, yang kedua nama dan yang terakhir asal.

|  |
| --- |
| #include <stdio.h>  void Biodata(, , ,){  int tahun\_sekarang = 2020;  // Lengkapi Function ini  }  int main() {  int tahunLahir;  char A[20], B[15];  scanf(" %d",&tahunLahir);  scanf(" %[^\n]%\*c",&A);  scanf(" %[^\n]%\*c",&B);  Biodata(tahunLahir, Namaku, Asal);  return 0;  } |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2001  Doni  Banjarmasin | Perkenalkan Nama Saya : **Doni**  Umur Saya : **19**  Saya Adalah Angkatan : **2020**  Asal Saya dari : **Banjarmasin** |
| 2003  Rina  Martapura | Perkenalkan Nama Saya : **Rina**  Umur Saya : **17**  Saya Adalah Angkatan : **2020**  Asal Saya dari : **Martapura** |

## 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>  void Biodata(int lahir, char nama[], char asal[])  {      //Lengkapi Function ini      int tahun\_Sekarang = 2020;      int umur = tahun\_Sekarang - lahir;      printf("Perkenalkan Nama Saya : %s", nama);      printf("\nUmur Saya : %d", umur);      printf("\nSaya Adalah Angkatan : %d", tahun\_Sekarang);      printf("\nAsal Saya dari : %s", asal);  }  int main()  {      int tahunLahir;      char Namaku[20], Asal[15];      scanf(" %d",&tahunLahir);      scanf(" %[^\n]%\*c",&Namaku);      scanf(" %[^\n]%\*c",&Asal);      Biodata(tahunLahir, Namaku, Asal);      return 0;  } |

Table 69 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | def Biodata(lahir, nama, asal):      #Lengkapi Function ini      tahun\_sekarang = 2020      umur = tahun\_sekarang - lahir      print(f"Perkenalkan Nama Saya : {nama}")      print(f"Umur Saya : {umur}")      print(f"Saya Adalah Angkatan : {tahun\_sekarang}")      print(f"Asal Saya dari : {asal}")  tahunLahir = int(input())  Namaku = input()  Asal = input()  Biodata(tahunLahir, Namaku, Asal)  :%.2d" % (jam, menit, detik)) |

Table 70 Source Code Bahasa Python Soal 5

## Output Program

Gambar 69 Screenshot Output Bahasa C Soal 5

3

Gambar 70 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Fungsi *void Biodata (int lahir, char nama[], char asal[])* untuk menyimpan sebuah karakter array.
* Pada baris 5: Fungsi *int tahun\_Sekarang = 2020;* menyatakan variabel tersebut dengan angka 2020.
* Pada baris 7-10: Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.
* Pada baris 12: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().

Pada baris 15: *char Namaku[20], Asal[15];* menyatakan dua tipe data array dengan maksimal 19 karakter dan 14 karakter

* Pada baris 16-21: Syntax *scanf (” ”)* berfungsi untuk menerima input. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *def Biodata(lahir, nama, asal):* menyatakan tiga variabel
* Pada baris 3: Fungsi *tahun\_sekarang = 2020* menyatakan variabel tersebut dengan angka 2020.
* Pada baris 5-8: Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# MODUL 6 : Array

# 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 |

## Source Code

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  2  13  14  15  16  17  18  19  20  21  22  23  24  25 | #include <stdio.h>  int main()  {      int a, b;      scanf("%d %d", &a, &b);      int matriks[a][b];      for(int i = 0; i < a; ++i)      {          for(int j = 0; j < b; ++j)          {              scanf("%d", &matriks[i][j]);          }      }      for(int i = 0; i < a; ++i)      {          for(int j = 0; j < b; ++j)          {              printf("%d ", matriks[i][j]);          }          printf("\n");      }      return 0;  } |

Table 71 Source Code Bahasa C Soal 1

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | a, b = map(int, input().split())  Bilangan = list(map(int, input().split()))  i = 0  while i < a \* b:      if i % b == 0 and i > 0:          print()      print(Bilangan[i], end=" ")      i+=1 |

Table 72 Source Code Bahasa Python Soal 1

## Output Program

Gambar 71 Screenshot Output Bahasa C Soal 1

Gambar 72 Screenshot Output Bahasa Python Soal 1

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int a, b;* menyatakan variabel *a, b* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: Fungsi *scanf("%d %d",&a ,&b);* Membaca input pengguna lalu menyimpannya dalam variabel *a, b.*
* Pada baris 6: *int matriks[a][b];* menyatakan matriks 2D dengan ukuran a x b berdasarkan nilai yang dimasukkan pengguna.
* Pada baris 8: Fungsi *for(int i = 0; i < a; ++i)* Memulai loop kedua yang berjalan dari 0 hingga a-1, yang digunakan untuk mengakses kolom matriks di dalam setiap baris.
* Pada baris 10: Fungsi *for(int i = 0; i < b; ++i)* Memulai loop kedua yang berjalan dari 0 hingga b-1, yang digunakan untuk mengakses kolom matriks di dalam setiap baris.
* Pada baris 12: *scanf("%d", &matriks[i][j]);* Membaca nilai integer dari input pengguna dan menyimpannya dalam elemen *matriks[i][j].*
* Pada baris 20-25: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *a, b =map(int, input().split()*membaca dua angka bilangan bulat dari input pengguna, memisahkannya berdasarkan spasi, mengubahnya menjadi bilangan bulat, dan menyimpannya ke dalam variabel a dan b.
* Pada baris 2: Fungsi *Bilangan = list(map(int, input().split()))* membaca input dari pengguna yang dipisahkan oleh spasi, mengubahnya menjadi bilangn bulat, dan menyimpannya ke dalam *list Bilangan*.
* Pada baris 4: Fungsi *while i < a \* b:* memulai loop *while* yang akan berjalan selama *i* kurang dari *a \* b*. Artinya, loop akan dijalankan sebanyak *a \* b* kali.
* Pada baris 5-8: Fungsi *if / else if / elif / else / while* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan

# SOAL 2

2. 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 |

## Source Code

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

Table 73 Source Code Bahasa C Soal 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | n = int(input())  b = list(map(int , input().split()))  i = 0  while i < n:      jumlah = b[i] \* (i+1)      print(jumlah, end=" ")      i +=1 |

Table 74 Source Code Bahasa Python Soal 2

## Output Program

Gambar 73 Screenshot Output Bahasa C Soal 2

****

Gambar 74 Screenshot Output Bahasa Python Soal 2

## Pembahasan

1. Pembahasan Bahasa C

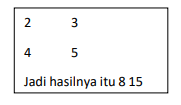
* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int n* menyatakan variabel *n* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *scanf(”&d, &n”);* membaca nilai yang akan dimamasukkan sebagai bilangan bulat lalu menyimpannya ke dalam variabel *n*.
* Pada baris 6: *int arr[n];* menyatakan array *arr* dengan variabel *n.*
* Pada baris 7: *for (int i = 1; i <= n;++i)* Memulai loop yang berjalan dari 0 hingga n-1, digunakan untuk mengisi array dengan nilai yang dimasukkan.
* Pada baris 9:  *scanf("%d", &arr[i]);* Membaca nilai bilangan bulat dari input pengguna dan menyimpannya dalam elemen *arr[i].*
* Pada baris 10-17: Pada baris 20-25: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *n = int(input())*membaca input dari pengguna, mengubahnya menjadi bilangan bulat, dan menyimpannya dalam variabel n.
* Pada baris 2: Fungsi *b = list(map(int , input().split()))*membaca input dari pengguna yang dipisahkan oleh spasi, mengubahnya menjadi bilangn bulat, dan menyimpannya ke dalam *list b*.
* Pada baris 4: Fungsi *while i < n:* memulai loop *while* yang akan berjalan selama *i* kurang dari *n*.
* Pada baris 5-8: Fungsi *if / else if / elif / else / while* blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan

# SOAL 3

3. 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 :



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 |

## 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 | #include <stdio.h>  int main()  {      int a, b;      scanf("%d %d", &a, &b);      if(a != b)      {          printf("Jumlah tidak sama");          return 1;      }      int arr1[a];      int arr2[b];      for(int i = 0; i < a; ++i)      {          scanf("%d", &arr1[i]);      }      for(int j = 0; j < b; ++j)      {          scanf("%d", &arr2[j]);      }      for(int i = 0; i < a; ++i)      {          int hasil = arr1[i] \* arr2[i];          printf("%d ", hasil);      }      return 0;  } |

Table 75 Source Code Bahasa C Soal 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | a, b = map(int, input().split())  if (a != b):      print("Jumlah tidak sama")      exit(1)  list\_a = list(map(int, input().split()))  list\_b = list(map(int, input().split()))  i = 0  while i < a:      jumlah = list\_a[i] \* list\_b[i]      print(jumlah, end=" ")      i+=1 |

Table 76 Source Code Bahasa Python Soal 3

## Output Program

Gambar 75 Screenshot Output Bahasa C Soal 3

3

Gambar 76 Screenshot Output Bahasa Python Soal 3

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4:: *int a, b;* menyatakan variabel *a* dan *b* sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: Fungsi *scanf("%d %d", &a ,&b);* Membaca bilangan bulat dari input pengguna lalu menyimpannya dalam variabel *a* dan *b*.
* Pada baris 6-7: *int arr1[a];* dan *int arr2 [b];* menyatakan dua array, *arr1* dengan variabel *a* dan *arr2* dengan variabel *b.*
* Pada baris 8-30: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *a, b = map(int, input().split())*membaca bilangan bulat dari input lalu memisahkannya dengan spasi, dan menyimpannya dalam variabel *n* dan *m*.
* Pada baris 5: *list\_a = list(map(int, input().split()))* membaca angka yang dipisahkan oleh spasi dari input pengguna, mengubahnya menjadi bilangan bulat, dan menyimpannya dalam *list\_a*.
* Pada baris 6: *list\_b = list(map(int, input().split()))* membaca angka yang dipisahkan oleh spasi dari input pengguna, mengubahnya menjadi bilangan bulat, dan menyimpannya dalam *list\_b*.
* Pada baris 10: *jumlah = list\_a[i] \* list\_b[i]* dalam setiap operasi loop, menghitung hasil perkalian elemen *list\_a[i]* dengan elemen *list\_b[i]* dan menyimpannya dalam variabel jumlah.
* Pada baris 9-12: Fungsi *if / else if / elif / else / while*  blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 4

4. 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 |

## 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  49  50  51  52  53 | #include <stdio.h>  #include <string.h>  int main()  {      char str1[1002];      char str2[1002];      char str3[1002];      fgets(str1, 1002, stdin);      fgets(str2, 1002, stdin);      int str1\_len = strlen(str1);      int str2\_len = strlen(str2);      if (str1\_len != str2\_len)      {          printf("Panjang kalimat berbeda, pesan palsu");          return 1;      }      int char\_sama = 0;      int char\_tidaksama = 0;      for(int i = 0; str1[i] != '\n'; ++i)      {          if(str1[i] == ' ' && str2[i] == ' ')          {              str3[i] = ' ';          }          else          {              if(str1[i] == str2[i])              {                  str3[i] = '\*';                  char\_sama++;              }              else if(str1[i] != str2[i])              {                  str3[i] = '#';                  char\_tidaksama++;              }          }          printf("%c", str3[i]);      }      printf("\n\* = %d", char\_sama);      printf("\n# = %d", char\_tidaksama);      if (char\_sama >= char\_tidaksama)      {          printf("\nPesan Asli");      }      else if(char\_sama < char\_tidaksama)      {          printf("\nPesan Palsu");      }      return 0;  } |

Table 77 Source Code Bahasa C Soal 4

|  |  |
| --- | --- |
| 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 | str1 = input()  str2 = input()  str3 = str1  if len(str1) != len(str2):      print("Panjang kalimat berbeda, pesan palsu")      exit(1)  i = 0  char\_sama = 0  char\_tidaksama = 0  while(i < len(str1)):      if str1[i] == ' ' and str2[i] == ' ':          str3 = str3[:i] +' '+str3[:i+1]      else:          if str1[i] == str2[i]:              str3 = str3[:i] +'\*'+str3[:i+1]              char\_sama+=1          elif str1[i] != str2[i]:              str3 = str3[:i] +'#'+str3[:i+1]              char\_tidaksama+=1      print(str3[i],end="")      i+=1  print()  print(f"\* = {char\_sama}")  print(f"# = {char\_tidaksama}")  if char\_sama >= char\_tidaksama:      print("Pesan Asli")  elif char\_sama < char\_tidaksama:      print("Pesan Palsu") |

Table 78 Source Code Bahasa Python Soal 4

## Output Program

Gambar 77 Screenshot Output Bahasa C Soal 4

3

Gambar 78 Screenshot Output Bahasa Python Soal 4

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *#include <string.h>* digunakan untuk mengimpor string yang diperlukan untuk fungsi-fungsi manipulasi string seperti strlen.
* Pada baris 3: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai integer. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 5-7: menyatakan tiga array karakter (string) dengan ukuran 1002 elemen masing-masing. *str1* dan *str2* akan menyimpan input dari pengguna, sementara *str3* akan menyimpan hasil komparasi.
* Pada baris 8-9: Membaca dua input dari pengguna dan menyimpannya dalam *str1* dan *str2*. Fungsi *fgets* digunakan untuk membaca string termasuk spasi hingga newline atau hingga 1001 karakter.
* Pada baris 10-11: Membaca dua baris input dari pengguna dan menyimpannya dalam *str1* dan *str2*. Fungsi fgets digunakan untuk membaca string termasuk spasi hingga newline atau hingga 1001 karakter.
* Pada baris 12-53: Syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. . Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *if / else if / elif / else / while*  blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1-3: *str1* dan *str2* akan menyimpan input dari pengguna, sementara *str3* akan menyimpan hasil komparasi.

Pada baris 12: Fungsi *while(i < len(str1)):* memulai loop *while* yang berjalan selama *i* kurang dari panjang str1.

* Pada baris 13-30: Fungsi *if / else if / elif / else / while*  blok pengondisian mengevaluasi nilai yang dimasukkan untuk menentukan kategori huruf yang sesuai. Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# SOAL 5

5. 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 |

## 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>  int main()  {      int n;      scanf("%d", &n);      int matriks1[n][n];      int matriks2[n][n];      int matriks3[n][n];      printf("Matriks A\n");      for(int i = 0; i < n; ++i)      {          for(int j = 0; j < n; ++j)          {              scanf("%d", &matriks1[i][j]);          }      }      printf("Matriks B\n");      for(int i = 0; i < n; ++i)      {          for(int j = 0; j < n; ++j)          {              scanf("%d", &matriks2[i][j]);          }      }      int sum  = 0;      for(int i = 0; i < n; ++i)      {          for(int j = 0; j < n; ++j)          {              for(int k = 0; k < n; ++k)              {                  sum+= matriks1[i][k] \* matriks2[k][j];              }              matriks3[i][j] = sum;              sum = 0;          }      }      printf("Matriks AXB\n");      for(int i = 0; i < n; ++i)      {          for(int j = 0; j < n ; ++j)          {              printf("%d ", matriks3[i][j]);          }          printf("\n");      }      return 0;  } |

Table 79 Source Code Bahasa C Soal 5

|  |  |
| --- | --- |
| 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 | n = int(input())  Matriks\_A = []  Matriks\_B = []  Matriks\_C = []  print("Matriks A")  for i in range(n):      baris = list(map(int, input().split()))      Matriks\_A.append(baris)  print("Matriks B")  for i in range(n):      baris = list(map(int, input().split()))      Matriks\_B.append(baris)  for i in range(n):      Matriks\_C.append([0] \* n)  for i in range(n):      for j in range(n):          for k in range(n):              Matriks\_C[i][j] += Matriks\_A[i][k] \* Matriks\_B[k][j]  print("Matriks AXB")  for i in range(n):      for j in range(n):          print(Matriks\_C[i][j], end=" ")      print() |

Table 80 Source Code Bahasa Python Soal 5

## Output Program

Gambar 79 Screenshot Output Bahasa C Soal 5

3

Gambar 80 Screenshot Output Bahasa Python Soal 5

## Pembahasan

1. Pembahasan Bahasa C

* Pada baris 1: Syntax *#include <stdio.h>* digunakan untuk mengimpor *library* standart input output agar bisa menggunakan perintah seperti *printf()* dan *scanf().*
* Pada baris 2: Syntax *main()* berfungsi untuk pintu masuk sebuah program. Maksud int di depan main adalah tipe data yang akan dikembalikan, artinya fungsi ini akan mengembalikan nilai sebagai bilangan bulat. Syntax yang ada di antara {...} merupakan bagian dari fungsi main().
* Pada baris 4: *int n;* menyatakan variabel  *n*  sebagai bilangan bulat yang akan digunakan untuk menyimpan data.
* Pada baris 5: *scanf(”&d, &n”);* membaca nilai yang akan dimasukkan sebagai bilangan bulat lalu menyimpannya ke dalam variabel m, *n*.
* Pada baris 6-8: Menyatakan tiga matriks 2D dengan ukuran *n x n*. *matriks1* dan *matriks2* akan diisi dengan nilai dari input pengguna, sementara *matriks3* akan menyimpan hasil perkalian matriks.
* Pada baris 9-48: syntax *printf (“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan. Syntax *(\n)* atau newline yang berfungsi memberikan baris baru pada kalimat yang akan dikeluarkan. Fungsi *(%d)* adalah format spesifier yang menunjukkan bahwa nilai yang ditampilkan adalah bilangan bulat. Fungsi *( ; )* atau titik koma memiliki fungsi sebagai penanda akhir pernyataan. Fungsi *(return 0; )* menandakan bahwa program telah selesai dan tidak mengalami kesalahan saat dieksekusi.

1. Pembahasan Bahasa Python

* Pada baris 1: Fungsi *n = int (input())*membaca bilangan bulat dari input lalu memisahkannya dengan spasi, dan menyimpannya dalam variabel *n*.
* Pada baris 2-4: tiga daftar kosong untuk menyimpan nilai-nilai matriks *Matriks\_A, Matriks\_B, dan Matriks\_C.*

Pada baris [7][12][15][18][25]: Fungsi *for i in range(n):* Menggunakan loop for untuk membaca n baris input dari pengguna.

* Pada baris 6: Syntax *print(“ “)* adalah input yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

# RANGKUMAN

**Modul 0 : How To Program**

Modul 0 : How To Program ini mempelajari syntax sederhana seperti *printf (” ”)* untuk bahasa C dan *print(” ”)* untuk bahasa Pyhton. Fungsi syntax printf(“ “) adalah untuk menginput teks yang akan dikeluarkan oleh output. Fungsi tanda *(“ “)* digunakan untuk mengapit teks yang akan ditampilkan.

**Modul 1 : Variabel, Tipe Data dan Operator**

Variabel adalah tempat penyimpanan yang digunakan untuk menyimpan data atau nilai yang dapat berubah selama program berjalan. Variabel diberi nama untuk mengidentifikasi data yang disimpannya. Contoh yang saya gunakan pada Modul 2 ini seperti :

*int x = 5;*

*int y = 7;*

*int z = 9;*

untuk bahasa C

x = 5

y = 7

z = 9

Untuk Bahasa Pyhton.

Tipe data adalah jenis nilai yang bisa disimpan dalam suatu variabel. Ada berbagai tipe data di C dan Python yang saya gunakan pada Modul 2 ini yaitu, *Integer (int)* untuk bilangan bulat dan *float* untuk bilangan desimal. Ada beberapa tipe data lain yang saya gunakan pada Modul lain yaitu seperti *Character (char)* dan String.

Operator digunakan untuk melakukan operasi pada variabel dan nilai. Ada berbagai jenis operator yaitu:

Operator Aritmatika : + , - , \* (kali), / (bagi), % (modulus)

Operator Pembanding: == (sama dengan), != (tidak sama dengan), > (lebih besar dari), < (lebih kecil dari), >= (lebih besar atau sama dengan), <= (lebih kecil atau sama dengan)

Operator Logika: && (AND), || (OR), ! (NOT) untuk bahasa C dan and, or, not untuk bahsa Pyhton.

Modul 2 : Input dan Output

Pada Bahasa C biasanya menggunakan fungsi *scanf* untuk membaca input dari pengguna. Fungsi ini memungkinkan kita untuk mengambil berbagai tipe data, seperti *integer, float, karakter, dan string.*

Contoh: scanf("%g", &nilaiPertama);

Pada Pyhton, fungsi *input* digunakan untuk membaca input dari pengguna. Fungsi ini mengembalikan input sebagai string.

Contoh: nama   = input("Nama                   : ")

**Modul 3 : Kondisional**

Pada Bahasa C, struktur kondisional menggunakan pernyataan if, else if, else, dan switch. *If, Else is, else, Switch* adalah pernyataan if digunakan untuk mengeksekusi blok kode jika suatu kondisi benar. *else if dan else* digunakan untuk menangani kondisi tambahan.

Contoh:  *if(a>b && a>c)*

Sedangkan pada Bahasa Pyhton menggunakan *If, elif, else.*

Contoh: *elif a>=c>=b : print(f"{b} {c} {a}")*

**Modul 4 : Loop**

Dalam bahasa C dan Pyhton Loop digunakan untuk menjalankan blok kode sejumlah tertentu. Biasanya terdiri dari tiga bagian: inisialisasi, kondisi, dan peningkatan/decrement. Dalam modul ini saya menggunakan 2 macam Loop yaitu For Loop dan While Loop.

Contoh Bahasa C:

* *for(int i = 1; i <= 50; ++i)*
* *while (i < m && j > n)*

Contoh Bahasa Pyhton:

* *for x in range(1, 51):*
* *while i < m and j > n:*

**Modul 5 : Fungsi**

Fungsi (function) adalah blok kode yang dapat dijalankan untuk melakukan instruksi tertentu.

Contoh Bahasa C:

 //Lengkapi Function ini

    return abs(nilai1 - nilai2);

Pada Python, fungsi didefinisikan dengan menggunakan kata kunci *def.*

Contoh Bahasa Pyhton:

*#Buatlah Function Disini*

*def MaxBilangan(a, b, c, d):*

**Modul 6 : Array**

Pada Bahasa C array didefinisikan dengan menentukan tipe data dari elemen array, diikuti oleh nama array, dan ukuran array di dalam tanda kurung siku [].

Contoh: *int arr1[a];*

Sedangkan pada Bahasa Python, tidak ada tipe data array bawaan seperti dalam Bahasa C. Namun, kita dapat menggunakan *list* untuk fungsi yang sama. List adalah struktur data yang serbaguna dan dinamis yang dapat menyimpan elemen dengan tipe data yang berbeda.

Contoh: *Bilangan = list(map(int, input().split()))*

# TAUTAN GIT

Berikut adalah tautan untuk source code yang telah dibuat.

https://github.com/username/praktikumpemrogramanweb1/tree/main/modul1