

KETENTUAN TUGAS PENDAHULUAN

- Tugas pendahuluan bertujuan untuk membantu mahasiswa dalam mempersiapkan diri sebelum mengikuti praktikum.
- Tugas pendahuluan bersifat tugas **INDIVIDUAL** dan **WAJIB** dikerjakan sebelum praktikum. Mahasiswa yang tidak mengerjakan akan memperoleh **PENALTI** pengurangan nilai praktikum sebesar 25%.
- Kerjakan soal dan susun jawaban dengan rapi dan jelas.
- Soal teori boleh cari dari internet atau modul dengan **MELAMPIRKAN SUMBER**. Jika sumber dari modul sertakan halaman, jika sumber dari internet sertakan link sumbernya (tidak boleh dari blogspot atau wordpress, harus dari web yang valid seperti codepolitan, w3schools dan lain-lain).
- Jawaban soal teori/non-koding **WAJIB DITULIS TANGAN**, lalu foto/scan jawaban tersebut dan **SCREENSHOT KODE PROGRAM** dan **OUTPUT/HASIL RUNNING PROGRAM** lalu masukan ke dalam file PDF dan **BERIKAN NOMOR DAN KETERANGAN CLASS SESUAI NOMOR JAWABAN DI PDFNYA** dengan format file **TP_ALPRO_MOD_KELAS_NIM.pdf**
- Jadikan **SEMUA JAWABAN PROGRAM ATAU FILE KODE PROGRAM DALAM 1 FILE ZIP** dengan format nama/ file **TP_ALPRO_MOD_KELAS_NIM.zip** **WAJIB DALAM ZIP**
- **SELURUH JAWABAN BAIK DI (SCREENSHOOT) PDF MAUPUN DI (PROGRAM/PROJECT) ZIP/RAR HARUS SAMA DAN SESUAI.**
- Upload file PDF dan ZIP ke google form yang dibuka pada waktu yang telah ditentukan
- Google Form akan dibuka sampai hari **KAMIS, 2 JUNI 2022, Jam 07.30 PAGI**
- **DILARANG KERAS MELAKUKAN SEGALA TINDAKAN PLAGIARISME**

- (Login dengan SSO, usernameigracias@student.telkomuniversity.ac.id)
- Video tutorial ALPRO dapat diakses di <https://tiny.cc/VideoTutorialALPRO>.

NB : JIKA MELANGGAR KETENTUAN DI ATAS

NILAI TP = 0

SEMANGATTT !!!

SOAL TUGAS PENDAHULUAN

Lengkapi setiap kode pseudocode di setiap soal-soal berikut ini ke dalam bahasa pemrograman

Go!

1. Diberikan tipe bentukan dan prosedur berikut

```
type Trec : < v1 : integer  
    vx : < v2, v3 : integer >  
    v4 : integer >  
  
procedure BanyakNilai(in/out rec: Trec)
```

Buat algoritma untuk mengolah nilai field dari parameter rec tersebut sehingga:

- a) field v1 menjadi berisi nilai terkecil dari semua nilai field,
- b) field v2 menjadi berisi jumlah dari semua nilai field semula,
- c) field v3 menjadi berisi rata-rata dari nilai-nilai semula dari field tersebut, dan
- d) field v4 menjadi berisi nilai terbesar dari semua nilai awal field.

2. procedure TambahData(in/out tab: array[1..N] of integer, in/out n: integer)

*{I.S. data masukan telah siap pada piranti masukan
F.S. array tab berisi sejumlah n bilangan bulat dari masukan, masukan berakhir apabila bilangan adalah 9999}*

3. function CariSekuensial(tab: array[1..N] of integer, v: integer) → integer
{Mengembalikan lokasi di mana nilai v berada di dalam array tab, atau -1 apabila v tidak ditemukan}
4. function NilaiMinimum(tab: array[1..N] of integer, n: integer) → integer
{Mengembalikan lokasi nilai minimum yang terdapat pada array tab yang berisi n bilangan bulat}
5. function NilaiRerata(tab: array[1..N] of integer, n: integer) → integer
{Mengembalikan nilai rata-rata dari seluruh bilangan yang terdapat pada array tab yang berisi n bilangan bulat}
6. procedure TerurutA(in/out tab : array[1..N] of integer, n: integer)

)

*{I.S. terdefinisi array tab yang berisi n bilangan bulat.
F.S. array tab terurut membesar menggunakan algoritma selection sort}*

7. procedure TerurutB(in/out tab : array[1..N] of integer, n:integer)
)

*{I.S. terdefinisi array tab yang berisi n bilangan bulat.
F.S. array tab terurut mengecil menggunakan algoritma insertion sort}*

8. function CariCepat(tab : array[1..N] of integer, n,v:integer) → integer
{Mengembalikan lokasi di mana nilai v berada di dalam array tab yang berisi n bilangan bulat dan terurut mengecil, atau -1 apabila v tidak ditemukan. Gunakan algoritma pencarian biner/belah tengah}
9. Baca procedure berikut ini dan apa yang dilakukan!

```
procedure Shaggy(tab : array[1..N] of Trec, n : integer)
  kamus
    i :integer
    found1,found2,found :Boolean
  algoritma
    found = false
    i = 2
    while i < n and not found do
      found1 = (tab[i-1].v1 == tab[i].vx.v2)
      found2 = (tab[i].vx.v3 == tab[i].v4)
      found = found1 and found2
      i = i + 1
    endwhile
    if found then
      output("Ada Shaggy disana. Yaitu ...?")
    endif
endprocedure
```