
Started on Friday, 16 April 2021, 12:41 PM

State Finished

Completed on Friday, 16 April 2021, 1:53 PM

Time taken 1 hour 11 mins

Marks 1100.00/1100.00

Grade **100.00** out of 100.00

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: lingkaran.py

Buatlah program yang digunakan untuk menghitung luas lingkaran. Program menerima input jari-jari (integer). Input jari-jari harus divalidasi sehingga selalu > 0 . Apabila jari-jari yang dimasukkan ≤ 0 , program akan mengeluarkan pesan error. Jika jari-jari yang dimasukkan valid, program menghasilkan luas lingkaran dengan rumus: $3.1415 * \text{jari-jari} * \text{jari-jari}$.

Keterangan:

Luas lingkaran harus ditulis dengan 2 angka di belakang koma.

Contoh input/output:

Input (data jari-jari)	Output
10	314.15
-1	Jari-jari harus > 0

Petunjuk:

Di Python 3, menuliskan beberapa angka di belakang koma dari bilangan real dapat dilakukan dengan cara menggunakan formatting spt. contoh berikut.

Contoh:

```
v = 38.483375
x = 12.566
print ("v is %.4f " % v + "while x is %.4f" % x)
```

akan mencetak sbb. di layar.

```
v is 38.4834 while x is 12.5660
```

Python 3

 [lingkaran.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.01 sec, 7.37 MB
2	25	Accepted	0.01 sec, 7.26 MB
3	25	Accepted	0.02 sec, 7.32 MB
4	25	Accepted	0.01 sec, 7.28 MB

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: bujursangkar.py

Buatlah program yang digunakan untuk menghitung luas bujur sangkar (persegi). Program menerima input sisi (dalam bentuk integer). Input sisi harus divalidasi sehingga selalu > 0 . Apabila sisi yang dimasukkan ≤ 0 , program akan mengeluarkan pesan error. Setelah sisi yang dimasukkan valid, program menghasilkan luas sisi dengan rumus: sisi * sisi

Contoh input/output:

Input	Output
10	100
-1	Sisi harus > 0

Python 3

 [bujursangkar.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.04 sec, 7.22 MB
2	25	Accepted	0.03 sec, 7.28 MB
3	25	Accepted	0.04 sec, 7.32 MB
4	25	Accepted	0.02 sec, 7.26 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Nama file: segitiga.py

Buatlah program yang digunakan untuk menghitung luas segitiga. Program menerima input alas dan tinggi (dalam bentuk integer). Input alas dan tinggi harus divalidasi sehingga selalu > 0 . Apabila alas dan tinggi yang dimasukkan ≤ 0 , program akan mengeluarkan pesan error. Setelah alas dan tinggi yang dimasukkan valid, program menghasilkan luas segitiga dengan rumus $0,5 * \text{alas} * \text{tinggi}$. Luas segitiga dibulatkan ke integer terdekat (gunakan fungsi **round**).

Contoh input/output:

Input (data alas dan tinggi)	Output
10 20	100
-1 20	Alas dan tinggi harus > 0

Python 3

 [segitiga.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.01 sec, 7.38 MB
2	20	Accepted	0.01 sec, 7.18 MB
3	20	Accepted	0.01 sec, 7.19 MB
4	20	Accepted	0.01 sec, 7.31 MB
5	20	Accepted	0.01 sec, 7.25 MB

Question **4**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: barang.py

Buatlah program yang digunakan untuk membaca masukan harga sejumlah barang yang dibeli di minimarket. Program pertama-tama akan membaca input N yang merupakan jumlah barang. N diasumsikan selalu valid ($N > 0$). Selanjutnya, program akan meminta input harga barang sebanyak N (input harga dianggap selalu valid) dan menghitung jumlah harga barang.

Contoh input/output:

Input	Output	Keterangan
3 2000 3000 1000	6000	N = 3 Total harga = 6000
5 7000 5000 4000 1000 2000	19000	N = 5 Total harga = 19000

Python 3

 [barang.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.17 sec, 7.36 MB
2	25	Accepted	0.08 sec, 7.16 MB
3	25	Accepted	0.07 sec, 7.33 MB
4	25	Accepted	0.06 sec, 7.27 MB

Question **5**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: tabungan.py

Buatlah program yang digunakan untuk membaca masukan jumlah tabungan dari sejumlah siswa di suatu kelas. Program pertama-tama akan membaca input N yang merupakan jumlah siswa. N diasumsikan selalu valid, yaitu $N > 0$. Selanjutnya, program akan meminta input nilai tabungan sebanyak N (input nilai tabungan dianggap selalu valid, yaitu > 0) dan menghitung jumlah tabungan.

Contoh input/output:

Input	Output	Keterangan
3 3000 5000 1000	9000	N = 3
5 7000 5000 4000 1000 3000	20000	N = 5

Python 3

 [tabungan.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.01 sec, 7.29 MB
2	25	Accepted	0.01 sec, 7.25 MB
3	25	Accepted	0.01 sec, 7.28 MB
4	25	Accepted	0.01 sec, 7.23 MB

Question **6**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: penduduk.py

Buatlah program yang digunakan untuk membaca masukan jumlah penduduk desa dari sebuah kecamatan di suatu kota. Program pertama-tama akan membaca input N yang merupakan jumlah desa dalam kecamatan. N diasumsikan selalu valid. Selanjutnya, program akan meminta input jumlah penduduk setiap desa sebanyak N (input jumlah penduduk desa dianggap selalu valid, yaitu > 0) dan menghitung jumlah penduduk kecamatan (total penduduk desa).

Contoh input/output:

Input	Output	Keterangan
3 3000 5000 1000	9000	N = 3
5 7000 5000 4000 1000 3000	20000	N = 5

Python 3

 [penduduk.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.02 sec, 7.23 MB
2	25	Accepted	0.02 sec, 7.25 MB
3	25	Accepted	0.01 sec, 7.19 MB
4	25	Accepted	0.02 sec, 7.33 MB

Question **7**

Correct

Mark 100.00 out
of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: nilai.py

Buatlah program yang digunakan untuk membaca masukan sejumlah nilai mahasiswa (dalam bentuk angka) untuk sebuah kelas mata kuliah, sampai pengguna mengetikkan -9999 (-9999 tidak termasuk nilai yang diproses). Harus diperiksa bahwa nilai yang dimasukkan berada dalam range 0-100. Jika ada data nilai yang salah, maka data tersebut diabaikan (tidak termasuk dalam pemrosesan).

Selanjutnya program menuliskan ada berapa banyak mahasiswa keseluruhan, berapa banyak yang lulus (yaitu yang mendapat nilai ≥ 40), berapa yang tidak lulus (nilai < 40), dan rata-rata nilai (tuliskan dengan 2 angka di belakang koma), untuk data-data yang valid. Jika tidak ada data yang valid, maka tuliskan: 0 (menunjukkan banyaknya mahasiswa).

Semua data nilai yang salah diabaikan (tidak termasuk yang diproses). Jika pengguna dari awal mengetikkan -9999 (artinya tidak ada data yang dimasukkan), tuliskan pesan: "Data nilai kosong".

Contoh input/output:

Input	Output	Keterangan
100 70 -70 80 30 101 -9999	4 3 1 70.00	2 data nilai salah, yaitu -70 dan 101 diabaikan Banyak mahasiswa = 4 Lulus = 3 Tidak lulus = 1 Rata-Rata = 70.00
100 80 70 100 -9999	4 4 0 87.50	Banyak mahasiswa = 4 Lulus = 4 Tidak lulus = 0 Rata-rata = 87.50
-100 800 -70 -100 -9999	0	Banyak mahasiswa = 0
-9999	Data nilai kosong	Tidak ada data yang dimasukkan

Python 3

 [nilai.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No.	Score	Verdict	Description
-----	-------	---------	-------------

No	Score	Verdict	Description
No	Score	Verdict	Description

1	8	Accepted	0.01 sec, 7.29 MB
2	8	Accepted	0.01 sec, 7.34 MB
3	8	Accepted	0.02 sec, 7.49 MB
4	8	Accepted	0.03 sec, 7.39 MB
5	8	Accepted	0.01 sec, 7.44 MB
6	8	Accepted	0.01 sec, 7.28 MB
7	8	Accepted	0.01 sec, 7.48 MB
8	8	Accepted	0.01 sec, 7.40 MB
9	9	Accepted	0.02 sec, 7.40 MB
10	9	Accepted	0.01 sec, 7.34 MB
11	9	Accepted	0.03 sec, 7.40 MB
12	9	Accepted	0.02 sec, 7.32 MB

Question **8**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: tinggi.py

Buatlah program yang digunakan untuk membaca masukan tinggi tubuh mahasiswa (dalam bentuk angka) dalam suatu kelas, sampai pengguna mengetikkan -999 (-999 tidak termasuk nilai yang diproses). Harus diperiksa bahwa nilai yang dimasukkan berada dalam range 100-300. Jika ada data nilai yang salah, maka data tersebut diabaikan (tidak termasuk dalam pemrosesan).

Selanjutnya program menuliskan ada berapa banyak mahasiswa keseluruhan, berapa banyak mahasiswa dengan tinggi ≤ 150 , berapa banyak mahasiswa dengan tinggi > 170 , dan berapa rata-rata tinggi mahasiswa. Tinggi rata-rata dituliskan tanpa koma (dibulatkan, gunakan fungsi **round**).

Semua data nilai yang salah diabaikan (tidak termasuk yang diproses). Jika pengguna dari awal mengetikkan -999 (artinya tidak ada data yang dimasukkan) atau tidak ada data yang valid, tuliskan pesan: "Data kosong".

Contoh input/output:

Input	Output	Keterangan
150 90 200 -90 -999	2 1 1 175	2 data tinggi mahasiswa salah, yaitu 90 dan -90 diabaikan Banyak mahasiswa = 2 Tinggi $\leq 150 = 1$ Tinggi $> 170 = 1$ Rata-rata = 175
150 160 170 -999	3 1 1 160	Banyak mahasiswa = 3 Tinggi $\leq 150 = 1$ Tinggi $> 170 = 1$ Rata-rata = 160
90 -999	Data kosong	Tidak ada data yang valid
-999	Data kosong	Tidak ada data

Python 3

 [tinggi.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	8	Accepted	0.12 sec, 7.23 MB
2	8	Accepted	0.07 sec, 7.43 MB
3	8	Accepted	0.15 sec, 7.34 MB

No	Score	Verdict	Description

5	8	Accepted	0.03 sec, 7.42 MB
6	8	Accepted	0.07 sec, 7.39 MB
7	8	Accepted	0.05 sec, 7.36 MB
8	8	Accepted	0.08 sec, 7.38 MB
9	9	Accepted	0.06 sec, 7.34 MB
10	9	Accepted	0.06 sec, 7.42 MB
11	9	Accepted	0.18 sec, 7.34 MB
12	9	Accepted	0.12 sec, 7.30 MB

Question 9

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: nilaiekstrim.py

Buatlah sebuah program yang digunakan untuk membaca sebuah integer N (asumsikan $0 < N \leq 100$).

Program kemudian membaca N buah integer dan menyimpan setiap integer ke dalam suatu array. Selanjutnya, program menerima masukan sebuah nilai integer, misalnya X, dan menampilkan:

- Jika X ada di array, apakah X adalah nilai maksimum (tuliskan "maksimum") atau nilai minimum (tuliskan "minimum") atau keduanya. Jika bukan nilai maksimum atau minimum, menuliskan "N#A".
- Jika X tidak ada di array, tuliskan "X tidak ada".

Contoh input/output:

Input	Output	Keterangan
6 1 -1 2 4 -1 2 -1	minimum	N = 6 Isi array = {1,-1,2,4,-1,2} X = -1 maksimum = 4 minimum = -1
5 3 2 2 4 2 3	N#A	N = 5 Isi array = {3,2,2,4,2} X = 3 maksimum = 4 minimum = 2
5 1 1 1 1 1 1	maksimum minimum	N = 5 Isi array = {1,1,1,1,1} X = 1 maksimum = 1 minimum = 1
6 1 -1 2 4 -1 2 8	8 tidak ada	N = 6 Isi array = {1,-1,2,4,-1,2} X = 8 X tidak ada dalam array

Python 3

 [nilaiekstrim.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.06 sec, 7.41 MB
2	10	Accepted	0.06 sec, 7.34 MB
3	10	Accepted	0.03 sec, 7.21 MB
4	10	Accepted	0.05 sec, 7.29 MB
5	10	Accepted	0.05 sec, 7.32 MB
6	10	Accepted	0.03 sec, 7.32 MB
7	10	Accepted	0.05 sec, 7.34 MB
8	10	Accepted	0.04 sec, 7.30 MB
9	10	Accepted	0.03 sec, 7.30 MB
10	5	Accepted	0.06 sec, 7.34 MB
11	5	Accepted	0.05 sec, 7.42 MB

Question **10**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: berat.py

Buatlah program yang digunakan untuk membaca masukan berat tubuh mahasiswa (dalam bentuk angka) dalam suatu kelas, sampai pengguna mengetikkan -999 (-999 tidak termasuk nilai yang diproses). Harus diperiksa bahwa nilai yang dimasukkan berada dalam range 30-200 (30 dan 200 termasuk). Jika ada data nilai yang salah, maka data tersebut diabaikan (tidak termasuk dalam pemrosesan).

Selanjutnya program menuliskan ada berapa banyak mahasiswa keseluruhan, berapa banyak mahasiswa dengan berat ≤ 50 , berapa banyak mahasiswa dengan berat ≥ 100 , dan berapa rata-rata berat mahasiswa. Berat rata-rata dituliskan tanpa koma (dibulatkan, gunakan fungsi **round**).

Semua data nilai yang salah diabaikan (tidak termasuk yang diproses). Jika pengguna dari awal mengetikkan -999 (artinya tidak ada data yang dimasukkan) atau tidak ada data yang valid, tuliskan pesan: "Data kosong".

Contoh input/output:

Input	Output	Keterangan
50 290 100 -90 -999	2 1 1 75	2 data berat mahasiswa salah, yaitu 290 dan -90 diabaikan Banyak mahasiswa = 2 Berat $\leq 50 = 1$ Berat $\geq 100 = 1$ Rata-rata = 75
150 40 60 -999	3 1 1 83	Banyak mahasiswa = 3 Berat $\leq 50 = 1$ Berat $\geq 100 = 1$ Rata-rata = 83
-999	Data kosong	Tidak ada data
-90 -999	Data kosong	Tidak ada data valid

Python 3

 [berat.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	8	Accepted	0.01 sec, 7.26 MB
2	8	Accepted	0.01 sec, 7.27 MB
3	8	Accepted	0.01 sec, 7.34 MB
4	8	Accepted	0.01 sec, 7.27 MB
5	8	Accepted	0.01 sec, 7.30 MB

5	8	Accepted	0.01 sec, 7.38 MB
No	Score	Verdict	Description

6	8	Accepted	0.01 sec, 7.19 MB
7	8	Accepted	0.01 sec, 7.35 MB
8	8	Accepted	0.01 sec, 7.32 MB
9	9	Accepted	0.01 sec, 7.37 MB
10	9	Accepted	0.01 sec, 7.37 MB
11	9	Accepted	0.01 sec, 7.38 MB
12	9	Accepted	0.01 sec, 7.27 MB

Question **11**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Nama file: caricharacter.py

Buatlah sebuah program yang digunakan untuk membaca sebuah integer positif N. Nilai N harus divalidasi sampai didapatkan nilai N yang benar, yaitu $0 < N \leq 100$. Jika masukan N salah, tuliskan pesan "Masukan salah. Ulangi!". Setelah didapatkan nilai N yang benar, program meminta masukan N buah character.

Selanjutnya, program menerima masukan sebuah character, misalnya CC, dan menghasilkan:

- Jika CC = 'S' atau CC = 's' (kependekan dari "small"), tuliskan urutan pertama ditemukannya character huruf kecil dan tuliskan hurufnya. Jika tidak ada character huruf kecil, tuliskan "Tidak ada huruf kecil".
- Jika CC = 'L' atau CC = 'l' (el, kependekan dari "large"), tuliskan urutan pertama ditemukannya character huruf besar dan tuliskan hurufnya. Jika tidak ada character huruf besar, tuliskan "Tidak ada huruf besar".
- Jika CC = 'X' atau CC = 'x', tuliskan urutan pertama ditemukannya character selain huruf dan tuliskan character-nya. Jika tidak ada character selain huruf, tuliskan "Semua huruf".
- Jika CC selain selain {'S','s','L','l','X','x'} tuliskan: "Tidak diproses".

Petunjuk:

- Simpan nilai-nilai character dalam array of character.
- Padanan standar ASCII untuk character huruf kecil 'a' s.d. 'z' adalah angka desimal 97 s.d. 122.
- Padanan standar ASCII untuk character huruf besar 'A' s.d. 'Z' adalah angka desimal 65 s.d. 90.

Contoh input/output:

Interaksi Input	Output	Keterangan
<u>-1</u> Masukan salah. Ulangi! <u>8</u> <u>A</u> <u>B</u> <u>a</u> <u>C</u> <u>%</u> <u>#</u> <u>Z</u> <u>*</u> <u>_</u> <u>S</u>	3 a	N = 8 (divalidasi 1 kali) CC = 'S' Character huruf kecil pertama muncul di urutan ke-3 yaitu 'a'
<u>3</u> <u>A</u> <u>B</u> <u>C</u> <u>L</u>	1 A	N = 3 CC = 'L' Character huruf besar pertama muncul di urutan pertama yaitu 'A'
<u>3</u> <u>a</u> <u>#</u> <u>\$</u> <u>L</u>	Tidak ada huruf besar	N = 3 CC = 'L' Tidak ada huruf besar
<u>3</u> <u>A</u> <u>4</u> <u>\$</u> <u>3</u>	Tidak diproses	N = 3 X = 99 Tidak ada yang diproses

Python 3

 [caricharacter.py](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	15	Accepted	0.01 sec, 7.30 MB
2	15	Accepted	0.01 sec, 7.29 MB
3	15	Accepted	0.01 sec, 7.38 MB
4	15	Accepted	0.01 sec, 7.32 MB
5	15	Accepted	0.01 sec, 7.29 MB
6	15	Accepted	0.01 sec, 7.33 MB
7	5	Accepted	0.02 sec, 7.35 MB
8	5	Accepted	0.05 sec, 7.39 MB

[◀ Praktikum 5 \(Shift 1\)](#)

[Dokumen Responsi 5 ▶](#)