

|              |                                     |
|--------------|-------------------------------------|
| Started on   | Monday, 12 September 2022, 11:02 PM |
| State        | Finished                            |
| Completed on | Monday, 12 September 2022, 11:38 PM |
| Time taken   | 35 mins 45 secs                     |
| Grade        | 600.00 out of 600.00 (100%)         |

Question **1**

Correct

Mark 100.00 out of 100.00

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Buatlah sebuah program bernama **cetakpersegi.c** yang mencetak sebuah persegi berdasarkan masukan n, dimana panjang dari sisi persegi adalah  $2n-1$  dan dibentuk dengan pola O dan X secara selang-seling.

Contoh Input 1

1

Contoh Output 1

O

Contoh Input 2

2

Contoh Output 2

OXO

XOX

OXO

Contoh Input 3

3

Contoh Output 3

OXOXO

XOXOX

OXOXO

XOXOX

OXOXO

**Perhatian: terdapat newline setelah keluaran output.**

C

 [cetakpersegi.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 20    | Accepted | 0.00 sec, 1.46 MB |
| 2  | 20    | Accepted | 0.00 sec, 1.57 MB |
| 3  | 20    | Accepted | 0.00 sec, 1.64 MB |
| 4  | 20    | Accepted | 0.00 sec, 1.48 MB |
| 5  | 20    | Accepted | 0.00 sec, 1.50 MB |

Question **2**

Correct

Mark 100.00 out of 100.00

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Diberikan dua buah string yang terdiri dari huruf kapital dan nonkapital. Tentukan apakah kedua string tersebut sama! Kedua string dikatakan sama apabila memiliki panjang yang sama dan huruf per hurufnya merupakan abjad yang sama.

Buatlah program dengan nama: eq\_string.c

Input:  
Dua buah string, panjang string dijamin kurang dari 100

Output:  
String Ya/Tidak

Contoh Input:  
aNimeWasnTaMistake  
animewasntamistake

Contoh Output:  
Ya

Penjelasan:  
Kedua string tersebut memiliki panjang dan abjad yang sama, sehingga merupakan string yang sama

Perhatian: terdapat newline setelah keluaran output

C

 [eq\\_string.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 20    | Accepted | 0.00 sec, 1.64 MB |
| 2  | 20    | Accepted | 0.00 sec, 1.58 MB |
| 3  | 20    | Accepted | 0.00 sec, 1.51 MB |
| 4  | 20    | Accepted | 0.00 sec, 1.53 MB |
| 5  | 20    | Accepted | 0.01 sec, 1.61 MB |

Question **3**

Correct

Mark 100.00 out of 100.00

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Diberikan sebuah input integer. Buatlah sebuah program yang menjumlahkan semua digit bilangan dalam integer tersebut

Buatlah program dengan nama: jumlah.c

Input:

Sebuah Integer

Output:

Jumlah setiap digit dari integer tersebut

Contoh Input:

12465

Contoh Output:

18

Penjelasan:

1+2+4+6+5=18

Note: terdapat newline setelah keluaran output

C

 [jumlah.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 20    | Accepted | 0.00 sec, 1.71 MB |
| 2  | 20    | Accepted | 0.00 sec, 1.71 MB |
| 3  | 20    | Accepted | 0.00 sec, 1.71 MB |
| 4  | 20    | Accepted | 0.00 sec, 1.61 MB |
| 5  | 20    | Accepted | 0.00 sec, 1.63 MB |

Question **4**

Correct

Mark 100.00 out of 100.00

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Buatlah sebuah program yang menerima satu bilangan integer positif dan memberikan output integer tersebut dengan tiap digitnya sudah dibalik.

Contoh input:  
12345

Contoh output:  
54321

Terdapat newline pada akhir output.  
Submit file **reverse.c**

C

 [reverse.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 20    | Accepted | 0.00 sec, 1.73 MB |
| 2  | 20    | Accepted | 0.00 sec, 1.55 MB |
| 3  | 20    | Accepted | 0.00 sec, 1.62 MB |
| 4  | 20    | Accepted | 0.00 sec, 1.59 MB |
| 5  | 20    | Accepted | 0.00 sec, 1.64 MB |

Question **5**

Correct

Mark 100.00 out of 100.00

|              |       |
|--------------|-------|
| Time limit   | 1 s   |
| Memory limit | 64 MB |

Kelas IF2111 Algoritma dan Struktur Data baru saja menyelesaikan praktikum pertama. Setelah praktikum selesai, salah satu mahasiswa penasaran dengan banyaknya jumlah siswa yang lulus, indeks rata-rata serta indeks akhir kelas. Oleh karena itu, Ia membuat sebuah program untuk mencari informasi tersebut.

Program akan menerima input berupa **sebuah integer secara berulang hingga menerima input berupa -999**. Lalu, program akan mengeluarkan tiga baris kalimat berupa "Jumlah mahasiswa yang lulus = ", "Nilai rata-rata = ", dan "Indeks akhir kelas = ".

Adapun beberapa aturan yang diterapkan adalah:

1. Nilai yang valid dan masuk kedalam perhitungan adalah nilai yang berada diantara range 0-4. Nilai yang berada diluar range tersebut akan dianggap data kotor dan tidak dimasukkan kedalam perhitungan.
2. Aturan indeks yang diterapkan adalah sebagai berikut
  1. Apabila nilai = 4.00, Indeks = 'A'
  2. Apabila nilai >= 3.00 dan < 4.00, Indeks = 'B'
  3. Apabila nilai >= 2.00 dan < 3.00, Indeks = 'C'
  4. Apabila nilai >= 1.00 dan < 2.00, Indeks = 'D'
  5. Apabila nilai >= 0.00 dan < 1.00, Indeks = 'E'
3. Minimal indeks untuk lulus adalah B (>=3.00)
4. Nilai rata-rata ditulis dengan 2 angka desimal
5. Terdapat new line di akhir kalimat
6. Apabila tidak ada data yang valid (dari semua masukan, tidak ada yang memenuhi aturan nomor 1), maka program akan mengembalikan "Tidak ada data"

Berikut adalah contoh input-output dari program

| Input                    | Output  |
|--------------------------|---|
| 1<br>2<br>3<br>4<br>-999 | Jumlah mahasiswa yang lulus = 2<br>Nilai rata-rata = 2.50<br>Indeks akhir kelas = C |
| 7<br>0<br>3<br>4<br>-999 | Jumlah mahasiswa yang lulus = 2<br>Nilai rata-rata = 2.33<br>Indeks akhir kelas = C |
| 4<br>4<br>4<br>4<br>-999 | Jumlah mahasiswa yang lulus = 4<br>Nilai rata-rata = 4.00<br>Indeks akhir kelas = A |
| -999                     | Tidak ada data  |

Submit file bernama **indeksalstrukdat.c**

C

 [indeksalstrukdat.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 10    | Accepted | 0.00 sec, 1.55 MB |
| 2  | 10    | Accepted | 0.00 sec, 1.73 MB |
| 3  | 10    | Accepted | 0.00 sec, 1.57 MB |
| 4  | 10    | Accepted | 0.00 sec, 1.46 MB |
| 5  | 10    | Accepted | 0.00 sec, 1.69 MB |
| 6  | 10    | Accepted | 0.00 sec, 1.68 MB |
| 7  | 10    | Accepted | 0.00 sec, 1.66 MB |
| 8  | 10    | Accepted | 0.00 sec, 1.68 MB |
| 9  | 10    | Accepted | 0.00 sec, 1.66 MB |
| 10 | 10    | Accepted | 0.00 sec, 1.55 MB |

Question **6**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Sebuah bilangan dapat disebut bilangan cantik bila terdapat angka 7 di bilangan tersebut atau bisa dibagi 7.

Contoh:

- 14 adalah bilangan cantik karena dapat dibagi 7
- 71 adalah bilangan cantik karena terdapat angka 7 di puluhan

Buatlah program sederhana dengan nama **bilangan.c**.

Pertama-tama program menerima input bilangan **N**.

**N** menandakan jumlah test case. Untuk setiap test case, program menerima input bilangan **M**.

Untuk menyelesaikan setiap test case, hitunglah jumlah total bilangan cantik dalam rentang 1 sampai **M**.

Contoh input:

3  
10  
20  
15

Contoh output:

7  
38  
21

Penjelasan:

Terdapat 3 test case.

| Test Case | M  | Hasil | Penjelasan  |
|-----------|----|-------|---|
| 1         | 10 | 7     | Bilangan cantik dari 1 sampai 10 cuma 7   |
| 2         | 20 | 38    | Bilangan cantik dari 1 sampai 20 adalah:<br>- 7<br>- 14<br>- 17<br>Jika dijumlahkan, hasilnya 38. |
| 3         | 15 | 21    | Bilangan cantik dari 1 sampai 15 adalah:<br>- 7<br>- 14<br>Jika dijumlahkan, hasilnya 21          |

Perhatian:

Anda bisa langsung menampilkan hasil ketika menerima test case baru.



Tidak perlu menunggu sampai menerima semua input test case.

Contoh keadaan terminal yang benar:

```
- Jika langsung mengeluarkan output ketika menerima test case
3      // N jumlah test case
10     // Test case pertama
7      // Output test case pertama
20     // Test case kedua
38     // Output test case kedua
15     // Test case ketiga
21     // Output test case ketiga

- Jika menunggu menerima semua test case baru mengeluarkan output.
3      // N jumlah test case
10     // Test case pertama
20     // Test case kedua
15     // Test case ketiga
7      // Output test case pertama
38     // Output test case kedua
21     // Output test case ketiga
```

C

 [bilangan.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

| No | Score | Verdict  | Description       |
|----|-------|----------|-------------------|
| 1  | 10    | Accepted | 0.00 sec, 1.49 MB |
| 2  | 10    | Accepted | 0.00 sec, 1.64 MB |
| 3  | 10    | Accepted | 0.00 sec, 1.71 MB |
| 4  | 10    | Accepted | 0.00 sec, 1.55 MB |
| 5  | 10    | Accepted | 0.00 sec, 1.60 MB |
| 6  | 10    | Accepted | 0.00 sec, 1.63 MB |
| 7  | 20    | Accepted | 0.00 sec, 1.60 MB |
| 8  | 20    | Accepted | 0.00 sec, 1.67 MB |

