# **Soal Final Praktikum**

**Richard**

1. Buat program untuk Print Hello World (poin 100)

**Fatwa (4 Easy) (2 Paket A 2 Paket B)**

1. Buat program yang memiliki 2 inputan, inputan pertama adalah angka, sebagai panjang array, dan inputan ke-dua adalah nilai dari arraynya, dari array tersebut cari jumlah dari nilai maksimal **(Poin 10)**

**Contoh Input 1:**

| **Panjang Array: 6**  **Nilai nilai Array: 5 2 4 5 2 1** |
| --- |

**Contoh Output 1:**

| **Nilai Maksimal adalah 5**  **Terdapat 2 nilai 5** |
| --- |

1. Buat program yang memiliki 2 inputan, inputan pertama adalah angka, sebagai panjang array, dan inputan ke-dua adalah nilai dari arraynya, buatlah fungsi yang akan menjumlahkan nilai ganjil baik positif maupun negatif dari array tersebut **(Poin 5)**  
   **Contoh Input 1:**

| **Panjang Array: 6**  **Nilai nilai Array: 1 3 4 7 2 -1** |
| --- |

**Contoh Output 1:**

| **Hasil dari 1 + 3 + 7 - 1 = 10** |
| --- |

1. Buatlah program yang menerima inputan angka, dan hitunglah jumlah bilangan prima dari 2, 3, …,n dimana n adalah angka yang diinput oleh user **(Poin 10)**  
   **Contoh Input 1:**

| **Panjang Angka: 10** |
| --- |

**Contoh Output 1:**

| **Terdapat 4 bilangan prima dari 2 sampai 10, yaitu: 2, 3, 5, 7** |
| --- |

1. Buatlah program untuk menyorting array yang di-input oleh user (**Poin 20)**  
   **Contoh Input 1:**

| **Input Array: 2 7 1 10 5 11 8 21** |
| --- |

**Contoh Output 1:**

| **Hasil sorting dari array adalah [1,2,5,7,8,10,11,21]** |
| --- |

**Haerul (1 Hard (Paket A) , 1 Intermediate (Paket B), 2 Easy (1 Paket A dan B)**

1. Program menerima satu inputan yang merupakan nilai n, cetak semua bilangan dari 1 sampai dalam n baris, dengan baris pertama terdiri dari satu bilangan, baris kedua terdiri dari dua bilangan, baris ketiga terdiri dari tiga bilangan, dan seterusnya sampai baris ke-n. (Easy)

**Contoh Input 1:**

| **5** |
| --- |

**Contoh Output 1:**

| **1**  **2 3**  **4 5 6**  **7 8 9 10**  **11 12 13 14 15** |
| --- |

1. Buatlah program untuk menyederhanakan pecahan (bentuk paling sederhana) dari pembagian dua buah bilangan yang diberikan (15 poin). (Paket B) (Easy)

**Contoh Input 1:**

| **6**  **10** |
| --- |

**Contoh Output 1:**

| **3/5** |
| --- |

1. Diberikan sebuah matriks persegi (n x n), hitung selisih absolut dari penjumlahan nilai diagonal kanan dan dan penjumlahan nilai diagonal kiri dari matriks tersebut. (20 poin) (Paket A)

**Format Input:**

Baris pertama merupakan ukuran matriks (n)

n baris selanjutnya adalah n buah bilangan yang merupakan isi dari matriksnya

**Format Output:**

Hasil selisih absolute dari penjumlahan nilai diagonal kanan dan dan penjumlahan nilai diagonal kiri dari matriks tersebut.

**Contoh Input:**

| **3 1 2 3 4 5 6 9 8 9** |
| --- |

**Contoh Output:**

| **2** |
| --- |

**Penjelasan Input/Output:**

Diagonal kanan = 1 + 5 + 9 = 15

Diagonal kiri = 3 + 5 + 9 = 17

Hasil = | 15 - 17 | = 2

1. Misalkan terdapat n kelompok bola berbeda warna, dengan setiap kelompok terdiri dari m bola. Kamu harus mengambil bola tersebut satu persatu dengan mata tertutup secara acak dan menyimpan bola-bola tersebut. Buatlah program untuk menentukan jumlah bola yang harus kamu ambil, agar kamu **pasti** akan memiliki minimal satu bola dari setiap kelompok. (30 poin) (Paket B)

**Format Input:**

Baris pertama adalah nilai n (jumlah kelompok warna bola).

Baris kedua adalah n buah bilangan yang menyatakan nilai m (jumlah bola) dari setiap kelompok bola.

**Format Output:**

Bilangan yang menyatakan jumlah bola yang harus kamu ambil

**Contoh Input 1:**

| **2 2 2** |
| --- |

**Contoh Output 1:**

| **3** |
| --- |

**Penjelasan Input/Output 1:**

n = 2, artinya, terdapat dua kelompok bola, sebut saja kelompok bola merah dan kelompok bola biru.

m1 m2 = 2 2, artinya, jumlah bola merah adalah 2 dan jumlah bola biru adalah 2.

* Jika jumlah bola yang diambil adalah 1, maka kombinasi bola yang mungkin didapatkan adalah:
  + satu bola merah atau,
  + satu bola biru

**Tidak mungkin** mendapatkan minimal satu bola dari setiap kelompok.

* Jika jumlah bola yang diambil adalah 2, maka kombinasi bola yang mungkin didapatkan adalah:
  + dua bola merah
  + dua bola biru
  + satu bola merah dan satu bola biru

**Tidak pasti** mendapatkan minimal satu bola dari setiap kelompok

* Jika jumlah bola yang diambil adalah 3, maka kombinasi bola yang mungkin didapatkan adalah:
  + dua bola merah dan satu bola biru
  + dua bola biru dan satu bola merah

**Pasti** mendapatkan minimal satu bola dari setiap kelompok.

**Contoh Input 2:**

| **3 2 3 4** |
| --- |

**Contoh Output 2:**

| **8** |
| --- |

**Mufti [1 Hard (Paket A), 1 Intermediate (Paket B), 2 Easy (Paket A dan B)]**

1. Diberikan lima bilangan bulat positif. Temukan nilai minimum dan maksimum yang dapat dihitung dengan menjumlahkan tepat empat dari lima bilangan bulat. Kemudian cetak nilai minimum dan maksimum masing-masing sebagai satu baris dari dua bilangan bulat panjang yang dipisahkan oleh spasi. **(Easy) (Poin 15) (Paket B)**

**Contoh Input 1:**

| 1 2 3 4 5 |
| --- |

**Contoh Output 1:**

| 10 14 |
| --- |

**Penjelasan:**

* 1. Penjumlahan 1: 2 + 3 + 4 + 5 = 14 (Max)
  2. Penjumlahan 2: 1 + 3 + 4 + 5 = 13
  3. Penjumlahan 3: 1 + 2 + 4 + 5 = 12
  4. Penjumlahan 4: 1 + 2 + 3 + 5 = 11
  5. Penjumlahan 5: 1 + 2 + 3 + 4 = 10 (Min)

**Contoh Input 2:**

| 7 69 2 221 8974 |
| --- |

**Contoh Output 2:**

| 299 9271 |
| --- |

1. Terdapat 2 mobil yang melaju dengan kecepatan masing-masing dan . Kedua mobil memulai start dengan jarak dan dimana dan . Buatlah program yang mengecek apakah kedua mobil tersebut **pasti** **akan bertemu di jarak tertentu**. Inputan terdiri dari dengan output “YES” jika kedua mobil pasti akan bertemu, “NO” jika kedua mobil tidak akan pernah bertemu. **(Hard) (Poin 35) (Paket A)**

**Contoh Input 1:**

| 0 3 4 2 |
| --- |

**Contoh Output 1:**

| YES |
| --- |

**Penjelasan:**

Mobil satu memulai di titik 0 dengan kecepatan 3

Mobil dua memulai di titik 4 dengan kecepatan 2

kecepatan artinya setiap detik bertambah sebanyak

Mobil 1: 0 - 3 - 6 - 9 - 12

Mobil 2: 4 - 6 - 8 - 10 - 12

Mereka bertemu saat di detik ke-4

**Contoh Input 2:**

| 0 2 5 3 |
| --- |

**Contoh Output 2:**

| NO |
| --- |

1. Buatlah program yang mencetak semua inputan angka. Jika angkanya merupakan bilangan desimal, maka angkanya akan dibulatkan ke bawah. Setiap inputan dipisahkan dengan tanda “-”. **(Easy) (Poin 10) (Paket A)**

**Contoh Input 1:**

| 1-2-dan-0.9-saya-1.3 |
| --- |

**Contoh Output 1:**

| 1 2 0 1 |
| --- |

**Contoh Input 2:**

| dan-92.3-100-@ |
| --- |

**Contoh Output 2**:

| 92 100 |
| --- |

1. Terdapat seorang asisten lab yang akan melakukan pertemuan dengan praktikannya untuk melakukan asistensi. Buatlah sebuah program untuk menentukan apakah asistensi tetap diadakan atau dibatalkan dengan syarat: **(Intermediate) (Poin 20) (Paket B)**

* Terdapat praktikan dengan praktikan minimal datang tepat waktu agar asistensi tetap diadakan.
* Terdapat waktu kedatangan , jika praktikan datang tepat waktu dan jika praktikan datang terlambat.

**Format Input:**

* 1. Baris pertama adalah , dimana dan
  2. Baris kedua adalah inputan list waktu kedatangan praktikan, dimana

**Format Output:**

Kembalikan dalam bentuk string “YES” jika asistensi dibatalkan, dan string “NO” jika asistensi tetap diadakan

**Contoh Input:**

| 4 3 -1 -3 2 1 |
| --- |

**Contoh Output:**

| YES |
| --- |

**Contoh Input 2:**

| 6 4 -20 8 0 1 -1 -5 |
| --- |

**Contoh Output 2:**

| NO |
| --- |

**Sony (1 Hard (Paket A), 2 Intermediate (Paket A dan B), 2 Easy (Paket A dan B))**

1. Buatlah sebuah program untuk mengekstrak username, domain dan ekstension dari sebuah email yang valid (Easy 15)

**Contoh Input:**

| **ssyukiao79@yahoo.com** |
| --- |

**Contoh Output:**

| **ssyukiao79**  **yahoo**  **com** |
| --- |

1. Seorang mahasiswa sedang melakukan observasi di sebuah terminal. Ia ingin mengetahui jenis mobil tertentu yang sering memasuki terminal tersebut.Sebelumnya ia telah mengelompokkan jenis-jenis mobil tersebut kedalam bentuk bilangan bulat mulai dari angka 1. Daftar mobil yang memasuki terminal tersebut diberikan dalam sebuah list yang berisi daftar bilangan bulat. Buatlah sebuah program untuk mengidentifikasi mobil mana yang paling sering memasuki sebuah terminal. Jika terdapat lebih dari 1 mobil dengan jumlah kemunculan terbanyak yang sama, kembalikan jenis mobil dengan id terendah. (Med 20)

**Contoh Input:**

| **1 4 4 4 5 3** |
| --- |

**Contoh Output:**

| **4** |
| --- |

1. Diketahui sebuah kata merupakan susunan beberapa karakter yang dapat terdiri dari huruf, angka dan tanda \_ (underscore). Susunan karakter tersebut harus dibatasi oleh karakter yang tidak terdapat pada sebuah kata (contohnya adalah spasi) atau dibatasi oleh batasan kiri atau batasan kanan dari sebuah kalimat (awal atau akhir kalimat). Misalkan diberikan sebuah N kalimat dan T kata. Buatlah sebuah program yang menerima beberapa inputan. Yang pertama adalah N yang merupakan jumlah kalimat yang akan digunakan. Setelah itu N baris berikutnya menyatakan masing masing kalimat yang akan diuji. Selanjutnya akan menerima inputan T yang merupakan jumlah kata yang akan diuji. Setelah itu T baris berikutnya menyatakan kata yang akan diuji. Output dari program ini adalah jumlah kemunculan setiap kata dalam seluruh kalimat yang dipisahkan dengan baris baru. (Med 25)

**Contoh Input:**

| **1**  **lunch day (lunchy( lunch .lunch. lunch123 1lunch45 +lunch=**  **1**  **lunch** |
| --- |

**Contoh Output:**

| **4** |
| --- |

1. Buatlah program untuk mengekstrak email dari sebuah teks. Tampilkan daftar email tersebut dalam keadaan terurut dipisahkan dengan tanda ; (Hard 30)

**Contoh Input:**

| **HackerRank is more than just a company**  **We are a tight group of hackers, bootstrappers, entrepreneurial thinkers and innovators. We are building an engaged community of problem solvers. Imagine the intelligence and value that a room would hold if it contained hackers/problem solvers from around the world? We're building this online.**  **Hypothesis: Every hacker loves a particular type of challenge presented in a certain set of difficulty. If we build a large collection of real world challenges in different domains with an engaging interface, it is going to be incredible! Join us to create history.**  **Available Positions**  **Product Hacker product@hackerrank.com**  **Challenge Curator**  **Product Evangelist**  **Product Designer**  **Content Creator**  **ACM World Finals Hacker**  **Backend C++ Hacker**  **Mail us at hackers@hackerrank.com to chat more. Or you can write to us at interviewstreet@hackerrank.com!**  **HACKERRANK PERKS**  **Working for a startup is hard work, but there are plenty of benefits of working for a small, fun, growing team.**  **[Image] Perk: Get tools for the jobAll the Right ToolsWe know that everyone's perfect workspace is unique to them. We will get you set up with whatever equipment you need to start hacking - a new 15” Macbook Pro or iMac, or a computer of your choice plus a display if you need it. Additionally, if you require any software or other tools, we've got it covered.[Image] Perk: Flexible HoursFlexible HoursBecause we work so hard, we encourage our employees to keep flexible hours and don't require them to track their time. A morning scrum and open communication ensures that the job gets done on time, and we rely on the honor system so that you can work on your own pace.[Image] Perk: HealthcareWellness SupportTo work hard, you have to be healthy. We will cover your health, dental, and visual insurance with no wait period. That means instant benefits from the day you're hired.[Image] Perk: Choice of LocationLocation, Location, LocationWe are the first Indian company to be backed by Y-Combinator, and as a result we have a thriving office in Bangalore and a growing office in Mountain View, CA. Depending on your residency or visa status, we will get you situated in one of our two offices, both of which are located in the heart of their country's tech industry.[Image] Perk: Choice of LocationCreative SupportIf you have a cool side project that you want to launch, we will pay for EC2/heroku servers to get it off the ground. Side projects fuel creativity and learning, which are crucial to the HackerRank culture.**  **CULTURE**  **The culture of a startup is reflective of the founders’ DNA. Larry Page & Sergey Brin were PhD’s from Stanford and that’s why Google is filled with high scoring graders from top schools and is very hard to get in if you’re not a CS major. Similarly, the hacker culture at Facebook is inspired by Zuckerberg, a hacker, the design culture by Steve Jobs and so on.**  **The adjective to describe the environment/founders here is relentless hardworkers. It might be a general trait of a startup but I’m pretty sure it’s a notch higher here and defines the culture. This is what has taken us this far. It’s not working in weekends or allnighters that count, but the effort that goes into building something intellectually engaging for hackers and making it fun is high.**  **You’ll have to embrace randomness and chaos. There’s some level of discipline (eg: daily scrums) but only so much. We push boundaries everyday, stretch our limits but no one complains because there’s a feeling of doing something great at the end of the day, every single day.** |
| --- |

**Contoh Output:**

| **hackers@hackerrank.com;interviewstreet@hackerrank.com;product@hackerrank.com** |
| --- |

1. Diberikan sebuah string yang merepresentasikan suatu tanggal pada kalender dengan format YYYYMMDD .Buatlah program untuk mengkonversi string tersebut menjadi format penanggalan Indonesia. Jika bagian harinya memiliki *leading zero*(diawali angka 0) maka tampilkan bagian harinya dalam 1 digit saja. (Easy 15)

**Contoh Input:**

| **19920619** |
| --- |

**Contoh Output:**

| **19 Juni 1992** |
| --- |

**← Ichsan →**

(1 Hard (Paket B), 3 Intermediate (1 Paket A dan 2 Paket B), 1 Easy (Paket A))

1. **[snake game]** Seekor ular berada pada koordinat (x,y). Batasi koordinat ular agar tetap berada dalam rentang 0-17 pada masing-masing sumbu (x dan y). Jika ular melebihi rentang tersebut maka akan kembali ke koordinat 0+offset posisi sebelumnya. **poin 15 (Paket A)**

**Contoh Input 1:**

| **12,21** |
| --- |

**Contoh Output 1:**

| **12,4** |
| --- |

**Contoh Input 2:**

| **-2,45** |
| --- |

**Contoh Output 2:**

| **15,11** |
| --- |

1. **[sorted alphabet]** Buatlah sebuah program yang dapat menyusun karakter sebuah string ke dalam list dengan urutan berdasarkan abjad (a-z). **poin 20 (paket B)**

**Contoh Input 1:**

| **hEsOyAm** |
| --- |

**Contoh Output 1:**

| **['A', 'E', 'h', 'm', 'O', 's', 'y']** |
| --- |

**Contoh Input 2:**

| **AeZaKmI** |
| --- |

**Contoh Output 2:**

| **['A', 'a', 'e', 'I', 'K', 'm', 'Z']** |
| --- |

1. **[X pattern]** Program menerima input n yang merupakan panjang baris dan kolom. Buatlah program untuk mencetak pola X menggunakan simbol **#** dengan ukuran n\*n. **poin 25 (paket A)**

**Contoh Input 1:**

| **5** |
| --- |

**Contoh Output 1:**

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

**Contoh Input 2:**

| **4** |
| --- |

**Contoh Output 2:**

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

1. **[the decryptor]** Sebuah pesan rahasia di enkripsi dengan cara ditulis secara terbalik kemudian di antara karakternya ditambahkan simbo**l $** . Kembalikan teks yang dienkripsi menjadi teks asli. **poin 30 (paket B)**

**Contoh Input 1:**

| **!$i$t$n$a$g$ $.$.$u$t$a$s$ $g$n$a$j$i$k** |
| --- |

**Contoh Output 1:**

| **kijang satu.. ganti!** |
| --- |

**Contoh Input 2:**

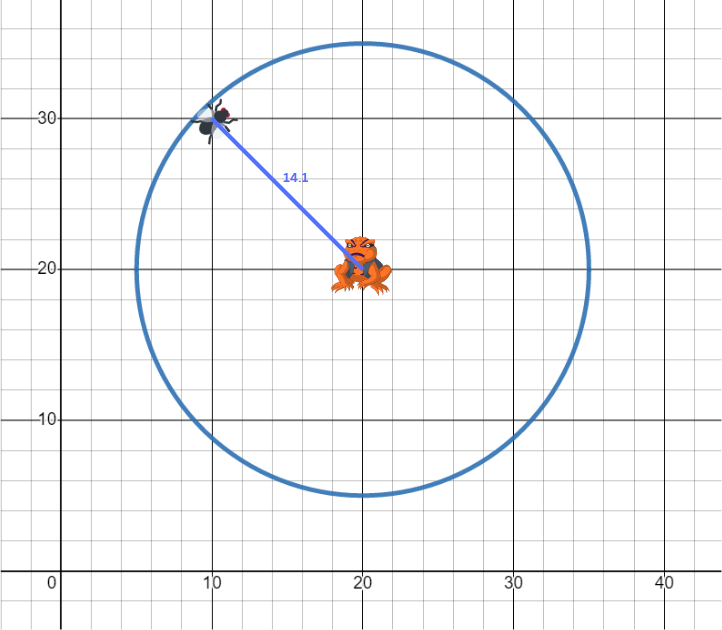
| **o$b$m$a$s$ $y$d$r$e$f** |
| --- |

**Contoh Output 2:**

| **ferdy sambo** |
| --- |

≤

1. **[frog’s pythagoras]** Seekor katak dapat menjangkau lalat pada radius 15. Periksa apakah seekor lalat berada dalam jangkauan si katak. Input pertama merupakan koordinat katak, kemudian input kedua merupakan koordinat lalat. **poin 35 (paket B)**

****

**Contoh Input 1:**

| **20,20**  **10,30** |
| --- |

**Contoh Output 1:**

| **dalam jangkauan** |
| --- |

**Contoh Input 2:**

| **20,20**  **10,8** |
| --- |

**Contoh Output 2:**

| **tidak dalam jangkauan** |
| --- |

**Richard (2 Hard (Paket A dan B), 1 Intermediate (Paket B), 3 Easy (1 Paket A dan 2 Paket B)**

1. **Mengelompokkan Kelereng**

Terdapat N buah kelereng dengan diameter tertentu, dan terdapat ember sebanyak K. Tugas Ikhsan adalah untuk mengelompokkan kelereng-kelereng tersebut pada ember-ember yang tersedia. Ikhsan yang kebingungan kemudian mempunyai ide brilian untuk mengelompokkan kelereng tersebut dengan menggunakan modulus, pada setiap kelereng, ia akan memasukkan kelereng pada ember ke [(diameter kelereng%K)+1]. Setiap ember dilabeli sesuai urutan abjad, ember ke-1 dilabeli 'A', ember ke-2 'B', dst.

Misalkan K=3, dan terdapat N=2 kelereng, kelereng pertama memiliki diameter 8, maka Ikhsan akan memasukkan kelereng tersebut ke ember [(8%3)+1] = 3 = ember C. Kelereng selanjutnya memiliki diameter sebesar 3, maka ia akan memasukkan kelereng tersebut ke ember [(3%3)+1] = 1 = ember A.

Maka Hasilnya:

| **Ember A: 1 kelereng**  **Ember B: 0 kelereng**  **Ember C: 1 kelereng** |
| --- |

Diberikan N buah kelereng dan K buah ember, berikan jumlah kelereng pada setiap ember.

**Constraint:**

| K = [1, 26] |
| --- |

**Contoh Input 1:**

| N = 2  K = 3  Dk = 8 3 |
| --- |

**Contoh Output 1:**

| **Ember A: 1 kelereng**  **Ember B: 0 kelereng**  **Ember C: 1 kelereng** |
| --- |

1. Fatwa yang kelelahan (easy)

Fatwa sedang joging di lapangan unhas, namun karena fisik Fatwa yang lemah, ia harus beristirahat setiap 35 langkah. Lapangan unhas memiliki keliling 500 meter. Fatwa ingin mengitari lapangan unhas sebanyak 1 kali. Pada 35 langkah pertama Fatwa berhasil melangkah sejauh 250 meter, pada langkah ke-70 Fatwa mencapai jarak 375 meter, lalu pada langkah ke-105 ia mencapai jarak 437.5 meter. Diberikan langkah keberapa, berikan sejauh mana Fatwa melangkah sekarang. Poin 10

**Constraint:**

| K = langkah % 35 |
| --- |

**Contoh Input 1:**

| 70 |
| --- |

**Contoh Output 1:**

| **375** |
| --- |

**Contoh Input 2:**

| 175 |
| --- |

**Contoh Output 2:**

| **484.375** |
| --- |

1. Lili senang bermain dengan bilangan bulat. Dia berhasil membuat sebuah permainan baru di mana ia menentukan sebuah bilangan dan kebalikannya. Contoh, diberikan nilai 12, kebalikannya adalah 21 dengan selisihnya 9. Bilangan 120 dibalik menjadi 21, dan kebalikannya adalah 99.

Dia memutuskan untuk menerapkan permainannya pada pengambilan keputusan. Dia akan melihat rentang hari dan hanya akan pergi ke bioskop pada hari yang indah. Diberikan rentang nilai hari, **[*i…j*]** dan nilai ***k***, tentukan hari dalam rentang nilai tersebut yang indah. Bilangan indah didefinisikan sebagai bilangan yang dapat dibagi habis oleh ***k***. Jika nilai hari adalah angka indah, maka hari itu adalah hari indah. Kembalikan jumlah hari indah dalam rentang tersebut. (Poin 15)

**Deskripsi Fungsi**

Lengkapi fungsi beautifulDays di editor di bawah ini. beautifulDays memiliki parameter seperti:

* int i: tanggal mulai
* int j: tanggal akhir
* int k: pembagi

**Returns**

* int: jumlah hari yang indah

**Format Input**

Sebuah baris dari tiga bilangan bulat yang dipisahkan oleh spasi yang menjelaskan nilai masing-masing dari ***i, j,*** dan ***k***

**Constraints**

**Sample Input**

| 20 23 6 |
| --- |

**Sample Output**

| 2 |
| --- |

1. [**https://www.beecrowd.com.br/judge/en/problems/view/1089**](https://www.beecrowd.com.br/judge/en/problems/view/1089) **(Hard)**

# **Soal Remedial Final Praktikum**

**Fatwa**

1. Buatlah program yang mem-print angka fibonacci yang dimulai dari 0 sampai n, n adalah angka yang di input oleh user yang lebih besar dari 3 dan lebih kecil dari 50,  
   3 < n < 50.   
   **Sample Input**

| 7 |
| --- |

**Sample Output**

| 0 1 1 2 3 5 8 |
| --- |

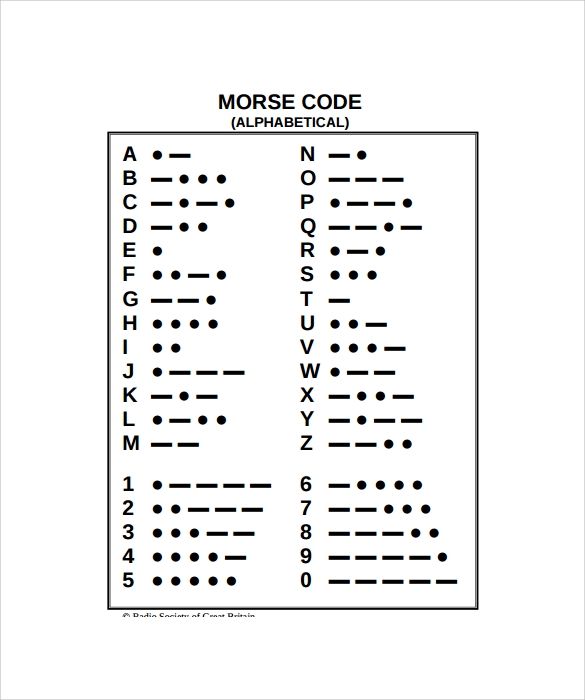
1. Buatlah program yang mem-print sequence angka dari 2 sampai n dimana angka index ke m adalah kelipatan dari angka index ke m-1

**Sample Input**

| 7 |
| --- |

**Sample Output**

| 2 4 8 16 32 64 128 |
| --- |

1. Buatlah program yang mentranslasikan kode morse, dimana (.) adalah pendek dan (-) adalah panjang, setiap kata dipisahkan oleh spasi (/)

**Sample Input**

| **.- -.. .- -.- .- .... / .--- .- .-- .- -... .- -.** |
| --- |

**Sample Output**

| adakah jawaban |
| --- |

1. Buatlah program yang mencari nilai duplikat dari dua buah list, dimana masing masing panjang list adalah n, dan 4 < n <= 10. Output dari program adalah: (Medium)  
   1. Jumlah duplikat yang ditemukan,

2. Index duplikat dari nilai tersebut di masing masing list

**Sample Input**

| **3 1 5 6 3 7**  **4 5 2 3 8 1** |
| --- |

**Sample Output**

| 2  3 pada list ke 1 index ke 0 dan list 2 index ke 3  1 pada list ke 1 index ke 1 dan list 2 index ke 6 |
| --- |

**Sony**

1. Diberikan sebuah inputan string S. Buatlah program untuk mengecek apakah inputan tersebut adalah sebuah nomor ponsel yang valid. Sebagai informasi contoh nomor ponsel yang valid adalah nomor tersebut terdiri 14 karakter yang diawali dengan +62 dan karakter yang tersisa merupakan sebuah digit bilangan. Tampilkan pesan “Valid” jika inputan tersebut adalah nomor ponsel yang valid dan “Tidak Valid” jika nomor ponsel tersebut tidak valid. (Easy 15)

**Contoh Input 1**

**Sample Input**

| +6285678995904 |
| --- |

**Sample Output**

| Valid |
| --- |

**Contoh Input 2**

**Sample Input**

| +62856789959047 |
| --- |

**Sample Output**

| Tidak Valid |
| --- |

1. Perusahaan NPX berinisiatif untuk membuat sistem password di perusahaannya untuk setiap pegawai. Hal ini dikarenakan sebagian besar pegawai di kantor tersebut sering lupa akan password yang mereka miliki. Oleh karena itu perusahaan NPX membuat suatu aturan dalam pembuatan password.

* Password harus memiliki panjang 12 karakter
* Password harus diawali oleh NPX dan diakhiri oleh NPX
* 6 karakter yang menyatakan hewan yang disukai oleh pegawai tersebut. Jika ternyata jumlah karakter untuk hewan tersebut tidak terdiri dari 6 karakter maka karakter tersisa akan diisi dengan angka 5

Tampilkan pesan “Valid” jika inputan tersebut adalah password yang valid dan “Tidak Valid” jika password tersebut tidak valid. (Easy 15)

NOTE:

Metode pembuatan password diatas bukanlah contoh implementasi yang tepat di dunia nyata. Contoh metode diatas hanyalah untuk menguji pemahaman anda tentang penggunaan regex

**Contoh Input 1**

**Sample Input**

| NPXayam55NPX |
| --- |

**Sample Output**

| Valid |
| --- |

**Contoh Input 2**

**Sample Input**

| NPayam5555NP |
| --- |

**Sample Output**

| Tidak Valid |
| --- |

1. Buatlah program untuk mengekstrak daftar email-email unik dari sebuah teks. Tampilkan jumlah email yang valid diikuti daftar email tersebut secara terurut (berdasarkan abjad) yang dipisahkan dengan baris baru. Untuk memudahkan anda, setiap username dalam email hanya akan memuat huruf alfabet lowercase, angka dan \_ (underscore). Selain itu, domain dan ekstensi yang valid adalah student.unhas.ac.id. (Medium 25)

Note:

Username pada email merupakan sekumpulan karakter yang berada pada sisi kiri tanda @. Contohnya adalah yukiao123 pada [yukiao123@gmail.com](mailto:yukiao123@gmail.com). Domain dan ekstensi adalah sekumpulan karakter yang berada pada sisi sebelah kanan tanda @. Pada contoh diatas adalah gmail.com.

**Contoh Input**

**Sample Input**

| Ucub dan Riko sedang menghukum praktikannya karena tidak mengumpulkan tugas tepat waktu. Mereka menyuruh setiap praktikan untuk menulis email mereka agar mereka dapat mengirimkan iklan produk mereka setiap waktu. Setelah itu, diperoleh daftar email yang masuk yakni [deril@student.unhas.ac.id](mailto:deril@student.unhas.ac.id), [iccang@student.unhas.ac.id](mailto:iccang@student.unhas.ac.id), [ical@student.unhas.ac.id](mailto:ical@student.unhas.ac.id), [alip@student.unhas.ac.id](mailto:alip@student.unhas.ac.id), dan yukiao@student.unhas.ac.com. |
| --- |

**Sample Output**

| **4**  [alip@student.unhas.ac.id](mailto:alip@student.unhas.ac.id)  [deril@student.unhas.ac.id](mailto:deril@student.unhas.ac.id)  [ical@student.unhas.ac.id](mailto:ical@student.unhas.ac.id)  iccang@student.unhas.ac |
| --- |

1. Buatlah program untuk mengecek apakah setiap kata dalam string diawali oleh huruf kapital. Tampilkan Yes jika setiap kata diawali oleh huruf kapital dan No jika terdapat kata yang tidak diawali oleh huruf kapital.

**Contoh Input 1**

**Sample Input**

| Praktikan Yang Kelelahan |
| --- |

**Sample Output**

| **Yes** |
| --- |

**Contoh Input 2**

**Sample Input**

| praktikaN yanG kelelahaN |
| --- |

**Sample Output**

| **No** |
| --- |

**Mufti:**

1. **(Pertemuan Tak Terduga).** Terdapat 2 orang sahabat yang saling berjumpa setelah 20 tahun. Buatlah Program yang membuat mereka melakukan percakapan sapaan kecil. **(Easy) (10 Point)**

Input Format:

Format terdiri dari satu baris yang berisikan 2 nama sahabat tersebut yang dipisahkan oleh “ , “ (Koma).

| Ihlasul Mufti Faqih , Ichsan Takwa |
| --- |

Constraint:

Inputan dapat memiliki spasi tidak terbatas di setiap kedua nama tersebut baik di sebelah kiri maupun kanan.

Format Output:

Output Terdiri dari 2 baris.

Baris pertama nama pertama akan menyapa nama kedua.

Baris kedua nama kedua akan menyapa nama pertama.

| Ihlasul Mufti Faqih: Halo, Selamat Pagi Ichsan Takwa Ichsan Takwa: Halo, Pagi Juga Ihlasul Mufti Faqih |
| --- |

1. **(Genap atau Ganjil).** Buatlah sebuah program yang akan memilih setiap angka yang diinput adalah genap atau ganjil. **(Easy) (15 Point)**

Format Input:

Inputan terdiri dari 2 baris.

Baris pertama adalah inputan string **ganjil** atau **genap.**

Baris kedua adalah inputan deretan angka yang dipisahkan oleh spasi.

| Ganjil  1 2 3 4 5 6 7 |
| --- |

Constraint:

Inputan string genap atau ganjil dapat dikombinasikan dengan uppercase atau lowercase. Contoh: “GeNAp”.

Format Output:

Output terdiri dari satu baris.

Program mengembalikan output dengan type list.

Output list genap jika yang diminta adalah genap, output list ganjil jika yang diminta ganjil. Output string **“Hanya Menerima Genap atau Ganjil”** jika yang diinput pada baris pertama bukan ganjil / genap.

| [1, 3, 5, 7] |
| --- |

1. **(Tebak Tipe Data).** Buatlah program yang menentukan tipe data dari data yang diinput apakah tipenya Integer, Float, String, atau Boolean.

Format Input:

Inputan terdiri satu baris list data untuk ditentukan tipe datanya.

| 123 **False** Halo **True** 0.2 |
| --- |

Constraint:

-

Format Output:

Program akan mencetak semua data beserta dengan tipe datanya.

| 123 = Integer **False** = Boolean Halo = String **True** = Boolean 0.2 = Float |
| --- |

1. **(Sulit Melupakan Rehan).** Alasan Intan penyanyi “Terlalu Sulit Lupakan Rehan” menyanyikan lagu tersebut dikarenakan dirinya telah disakiti oleh Rehan. Oleh karena itu dia semakin selektif dalam memilih cowok, dia tidak akan menerima cowok yang dalam namanya terdapat kata yang berawalan huruf **“R”** atau jumlah panjang namanya adalah ganjil.

Format Input:

Inputan terdiri satu baris sebuah nama.

| Gilang Rehan |
| --- |

Constraint:

Nama dapat terdiri dari beberapa kata. Spasi tidak akan dihitung sebagai panjang nama. Contoh:

Budi Sabdi memiliki panjang 9

Format Output:

Output terdiri dari satu baris **“BIG NO”** atau **“BIG YES”.**

**“BIG NO”** jika nama cowok tersebut tidak sesuai kriteria Intan, **“BIG YES”** jika nama tersebut sesuai kriteria Intan. Contoh:

Budi Sabdi BIG NO

Budi Rehana BIG NO

Budi Abdi BIG YES

| BIG NO |
| --- |

**← Ichsan →**

(2 Easy Paket Siang, 1 Easy Paket Pagi, 1 Medium Paket Pagi)

**1**. **@pagi** **[MABAR KUYY]** Group asistensi 12 yang beranggotakan Henokh, Fikry, dan Aditya sedang bermain game FPS. Tentukan MVP dan cetak masing” komentar yg diberikan tiap pemain dalam suatu round berdasarkan poin yang di input:

| **total poin** | **komentar** | **\*keterangan** |
| --- | --- | --- |
| 81 - ∞ | WP | well played |
| 41 - 80 | N1 | nice one |
| 0- 40 | MB | my bad |

**Pemain dengan poin tertinggi akan menjadi MVP**

Terdapat 3 inputan poin dengan urutan Henokh, Fikry, dan Aditya

| 25  57  25 |
| --- |

Terdapat 2 output dengan masing” pemain MVP dan komentar

| MVP : Fikry  WP |
| --- |

total poin : 25 + 57 + 25 = 107 (dalam rentang 81 - tak hingga) = **WP**

**2**. **@siang** **[Liga Dagelan]** Dua list berikut menyimpan data nama-nama klub yang akan bertanding di liga champion 2099

club\_A = ["R. Madrid", "Bayern" , "M. City" , "PSG" , "Persija" , "Arsenal" , "Ajax" , "Chelsea ", "Inter"]

club\_B = ["Barca" , "Dortmund", "M. United", "PSM Makassar", "Bhayangkara", "Liverpool", "Tottenham", "Juventus", "Atletico"]

Program menerima input nama salah satu tim, tentukan lawan dari tim tersebut dan urutan jadwal bertandingnya.

Tim pada klub A akan bertanding melawan tim klub B dengan posisi yang sama dalam list.

Urutan jadwal pertandingan juga berdasarkan posisi klub dalam list

contoh : pertandingan ke-1 adalah R.Madrid vs Barca

## [string] Input nama klub

| Bayern |
| --- |

## 

## [string] nama klub lawan

[integer] urutan jadwal pertandingan

| Dortmund  2 |
| --- |

## 

| PSM Makassar |
| --- |

| PSG  4 |
| --- |

**3.@siang** **[Final Dagelan]** Pada babak final Real Madrid akan melawan rivalnya PSM Makassar. Tentukan klub mana yang menang berdasarkan hasil adu penalti. Inputan pertama merupakan hasil tendangan penalti, dan inputan kedua adalah hasil klub PSM Makassar

Inputan berupa nilai-nilai **boolean** yang dipisah spasi

Output berupa **string** nama klub yang menang

Contoh Input:

| True False True True False  False True True True True |
| --- |

Contoh Output:

| PSM Makassar |
| --- |

**4. @pagi**  **[Pola Kotak]** Buatlah pola persegi menggunakan karakter “-” isi persegi, dan “#” bagian pinggir persegi dengan ukuran n\*n

inputan berupa Integer yang merupakan ukuran persegi

Contoh Input:

| 6 |
| --- |

pola berbentuk persegi dengan ukuran n\*n

Contoh Output:

| ######  #----#  #----#  #----#  #----#  ###### |
| --- |