

DOKUMEN

Salinan Soal Praktikum 2

Algoritma dan Pemrograman

Sistem dan Teknologi Informasi

Tahun 2024

Problem 1: Menemukan Bilangan Prima (Finding Prime Numbers)

Nama File: MisiKebin.c

Kebin diberikan sebuah misi dari Gro, yaitu membuat program untuk mencari banyak bilangan prima di antara dua bilangan. Sebenarnya, ini adalah misi yang mudah bagi Kebin karena ia sudah memahami konsep bilangan prima setelah belajar dengan Stewart. Namun, Gro memberikan program awal yang harus dipatuhi oleh Kebin. Sayangnya, program yang diberikan memiliki struktur prosedur yang kurang tepat.

Program awal yang diberikan oleh Gro:

[MisiKebin.c](#)

[x] Note: File yang diberikan sudah terselesaikan, tapi perintah pengerjaan masih ada berbentuk TODO yang ada dalam file.

Tugas Anda:

Perbaiki dan selesaikan program yang diberikan oleh Gro agar dapat berfungsi dengan benar.

Format Masukan:

- Program akan menerima dua integer A dan B dalam satu baris, dipisahkan oleh spasi.
- Batasan nilai : $1 \leq A \leq B \leq 10000$

Format Keluaran:

- Program harus mencetak satu bilangan bulat yang merupakan jumlah bilangan prima dalam rentang [A, B] **diakhiri dengan endline (“\n”)**

Contoh Masukan dan Keluaran:

No	Masukan	Keluaran	Keterangan
1.	10 20	4	Bilangan prima antara 10 dan 20 adalah 11, 13, 17, dan 19, total ada 4.
2.	2 10	4	Bilangan prima antara 2 dan 10 adalah 2, 3, 5, dan 7, total ada 4.

Problem 2: Pelacak Kematangan Pisang (Banana Ripeness Tracker)

Nama File: LaporanPisang.c

Pada suatu pagi yang cerah, Gro menerima kiriman spesial dari istrinya, Luly. Kiriman tersebut adalah sebuah keranjang besar berisi pisang segar. Namun ternyata pisang-pisang itu memiliki tingkat kematangan yang bervariasi. Ada yang masih mentah, ada yang sudah matang sempurna, dan ada pula yang terlalu matang.

Gro yang sangat teliti tidak ingin ada satupun pisang yang hilang, apalagi dimakan sembunyi-sembunyi oleh Toto. Oleh karena itu, Gro meminta Toto untuk memisahkan dan mencatat setiap pisang berdasarkan tingkat kematangannya. Tak hanya itu, Gro juga ingin mendapatkan laporan progress klasifikasi pisang setiap saat agar bisa memastikan Toto tidak mencuri pisang-pisang favoritnya.

Untuk membantu tugasnya, Toto memutuskan untuk membuat program sederhana yang dapat mencatat dan melaporkan tingkat kematangan setiap pisang dalam keranjang.

Fitur Program:

Berikut fitur program sederhana yang harus dibuat oleh Toto: 1. Memproses data pisang - Memasukkan jumlah hari setelah pisang dipetik - Program akan mengklasifikasikan pisang ke dalam kategori kematangan yang sesuai 2. Menampilkan jumlah pisang di setiap kategori 3. Program akan **terus dijalankan** hingga **seluruh pisang** telah diklasifikasi

Kategori Pisang:

Berikut kategori pemisahan pisang berdasarkan hari setelah pisang dipetik:

No	Jumlah hari pisang dipetik	Kategori
1.	0-2 hari	Mentah
2.	3-5 hari	Matang
3.	Lebih dari 5 hari	Terlalu matang

Format Masukan:

- Baris pertama: bilangan bulat positif **N** ($0 < N \leq 100$) yang menunjukkan banyaknya pisang dalam keranjang.

- Baris kedua dan seterusnya berisi perintah-perintah yang dapat dijalankan oleh program:
 - Masukkan angka 1 untuk memproses pisang
 - Setelah memasukkan angka 1, program meminta masukan hari **D ($D \geq 0$)** setelah pisang dipetik
 - Masukkan angka 2 untuk menampilkan laporan

Format Keluaran:

- Jika memilih **1**, tampilkan kategori kematangan pisang.
- Jika memilih **2**, tampilkan laporan jumlah pisang di setiap kategori.
- Jika memilih angka **bukan 1 atau 2**, tampilkan pesan “Input salah. Masukkan kembali input dengan benar.”
- Setelah seluruh pisang diproses, tampilkan pesan “Program selesai”

Contoh Masukan dan Keluaran:

No	Masukan	Keluaran	Keterangan
1.	5 1 1 1 4 2 1 7 4 1 5 1 0	Kategori: Mentah Kategori: Matang Laporan Kematangan Pisang Mentah: 1 Matang: 1 Terlalu matang: 0 Kategori: Terlalu matang Input salah. Masukkan kembali input dengan benar. Kategori: Matang Laporan Kematangan Pisang Mentah: 1 Matang: 2 Terlalu matang: 1 Kategori: Mentah Program selesai	

Pastikan setiap output diakhiri oleh **endline** (“\n”)!

Problem 3: Generator Kombinasi Password (Password Combination Generator)

Nama File: NimonsBruteforce.c

Nimons gagal menjalankan sebuah misi yang diberikan oleh Dr. Neroifa. Sebagai akibatnya, mereka terkurung di dalam sebuah ruangan dengan akses keluar ber-password. Untungnya, penjaga ruangan tersebut tanpa sengaja membuka akses ruangan tersebut dengan tangannya yang berminyak. Toto yang memiliki ide cemerlang pun menebak angka-angka yang digunakan sebagai password. Namun sayangnya, Toto tidak bisa mengingat urutan angka-angka tersebut.

Ternyata Kebin memiliki dokumen yang dapat membantu mencari seluruh kombinasi password dengan lebih mudah. Silakan gunakan [NimonsBruteforce.h](#) dan [driver.c](#). Karena dokumen tersebut ternyata adalah dokumen rahasia Dr. Neroifa, maka kalian **tidak diperkenankan untuk merubah isi dokumen** tersebut sedikit pun karena Kebin takut bahwa informasi penting di dokumen tersebut hilang.

Bantulah para Nimons untuk melakukan bruteforce terhadap angka-angka pada panel password tersebut.

Format Masukan:

- Satu buah bilangan $0 < N \leq 10^9$, dengan masing-masing digit tidak akan ada angka yang sama

Format Keluaran:

- semua susunan kemungkinan password yang dipisahkan oleh new line

Contoh Masukan dan Keluaran:

No	Masukan	Keluaran
1.	12	12 21
2.	123	321 312 231 213 132 123

No	Masukan	Keluaran
3.	2314	4132 4123 4312 4321 4213 4231 1432 1423 1342 1324 1243 1234 3412 3421 3142 3124 3241 3214 2413 2431 2143 2134 2341 2314

Notes:

- Pastikan setiap output diakhiri oleh **endline** (“\n”)!
- Kumpulkan hanya file **NimonsBruteforce.c**

Problem 4: Mesin Waktu (Time Machine)

Nama File: JelajahWaktu.c

Gro sedang merencanakan misi besar untuk mencuri bulan, tetapi dia menyadari bahwa perhitungannya salah! Untuk memperbaikinya, dia meminta bantuan Dr. Nefario untuk membuat sebuah mesin waktu sederhana yang dapat menghitung tanggal setelah beberapa hari ke depan.

Dr. Nefario pun berhasil menciptakan sebuah mesin yang menerima input berupa tanggal dalam format "dd MMMM yyyy" (contoh: 5 Januari 2025) dan sebuah angka n , yang menunjukkan berapa hari ke depan Gro ingin melakukan aksinya. Mesin ini akan menghitung tanggal yang tepat setelah n hari dari tanggal awal.

Lengkapi fungsi - fungsi pada file [JelajahWaktu.c](#) dan kumpulkan ulang pada soal ini.

[x] Note: File yang diberikan sudah terselesaikan, tapi perintah pengerjaan masih ada berbentuk @brief yang ada dalam file.

Format Masukan:

- Satu buah tanggal dalam format "dd MMMM yyyy" (contoh: 5 Januari 2025)
- Satu buah bilangan $1 \leq n \leq 10^6$ yang menunjukkan berapa hari ke depan Gro akan melakukan aksinya

Format Keluaran:

- Tanggal setelah n hari dari tanggal awal

Contoh input dan output:

No	Masukan	Keluaran	Keterangan
1.	20 Maret 2025 365	Jumat, 20 Maret 2026	
2.	20 Maret 2025 1	Jumat, 21 Maret 2025	
3.	1 Januari 1 13522531	Selasa, 8 Juni 37024	

Problem 5: Bilangan Bingro (Bingro Number)

Nama File: MisiKebin2.c

Gro kembali memberikan misi yang lebih sulit kepada Kebin! Kali ini, Kebin diminta untuk mencari **banyak bilangan prima** dalam rentang [A,B], tetapi dengan aturan tambahan:

1. Setiap bilangan prima yang ditemukan, jumlah semua digitnya juga harus dihitung dan diperiksa apakah prima atau tidak.
2. Jika jumlah digitnya juga merupakan bilangan prima, maka bilangan tersebut tetap dihitung dalam total, jika tidak maka tidak dihitung.
3. Setelah mendapatkan total bilangan prima yang memenuhi syarat, nilai tersebut harus dikonversi menjadi "Bilangan Bingro".

Gunakan jawaban dari misi sebelumnya (soal nomor 1) dan buatlah **dua buah fungsi** tambahan untuk **menghitung jumlah digit suatu bilangan** dan **mengonversi integer ke Bilangan Bingro**.

Definisi Bilangan Bingro:

Bilangan Bingro adalah representasi biner terbalik dari suatu bilangan, tetapi tanpa leading zero.

Contoh: - 16 → Biner: 10000 → Bingro: 1 - 19 → Biner: 10011 → Bingro: 11001

Format Masukan:

- Program akan menerima dua integer A dan B dalam satu baris, dipisahkan oleh spasi.
- Batasan nilai : $1 \leq A \leq B \leq 10000$

Format Keluaran:

- Program harus mencetak Bilangan Bingro yang merupakan total jumlah bilangan prima yang memenuhi syarat **diakhiri dengan endline ("\\n")**

Contoh Masukan dan Keluaran:

No	Masukan	Keluaran	Keterangan
1.	1 50	101	Bilangan prima yang memenuhi syarat: 2, 3, 5, 7, 11, 23, 29, 41, 43, 47. Total = 10. Bilangan Biner dari 10 adalah 1010. Sehingga Bilangan Bingro dari 10 adalah 101

Hint Konversi 19 ke biner: $19 \div 2 = 9$, sisa 1

$9 \div 2 = 4$, sisa 1

$4 \div 2 = 2$, sisa 0

$2 \div 2 = 1$, sisa 0

$1 \div 2 = 0$, sisa 1

Hasil biner: 10011