

dilakukan pembersihan (maintenance) berkala.

- **Why 5:** Mengapa tidak pernah dilakukan pembersihan berkala? **[AKAR MASALAH]:** Karena laboratorium komputer sekolah belum memiliki Standard Operating Procedure (SOP) tertulis mengenai jadwal perawatan berkala untuk aset komputer.

Perhatikan contoh di atas! Jika guru hanya melihat Why 1, solusinya mungkin hanya menghukum atau memarahi siswa. Namun, berkat berpikir komputasional dan 5 Whys, kita tahu bahwa solusi sejatinya adalah membuat aturan SOP perawatan komputer laboratorium. Masalah selesai sampai ke akarnya!

LAMPIRAN 2: LEMBAR KERJA PESERTA DIDIK (LKPD)

LEMBAR KERJA DETEKTIF BERPIKIR KOMPUTASIONAL

SMK MUHAMMADIYAH 1 CILACAP

Nama Kelompok	:	Kelas / Semester	: X SMK / Ganjil
Anggota Tim	1. 2.	3. 4.	

LANGKAH 1: ORIENTASI PADA MASALAH KONTEKSTUAL

STUDI KASUS RIIL:

Setiap kali bel istirahat pertama berbunyi (pukul 10.00 WIB), area kantin utama SMK Muhammadiyah 1 Cilacap selalu mengalami penumpukan dan antrean siswa yang berdesakan parah. Banyak siswa kelas X yang akhirnya tidak kebagian makanan atau kehabisan waktu istirahat hanya untuk mengantre. Akibatnya, mereka kembali ke kelas dengan perut kosong, lemas, dan konsentrasi belajar mereka pada jam pelajaran Koding setelah istirahat menurun drastis.

LANGKAH 2: DEKOMPOSISI (MEMECAH MASALAH BESAR)

Gunakan pisau analisa Dekomposisi kelompokmu! Pecahlah masalah besar "Antrean Kantin Parah" di atas menjadi minimal 3 elemen/sub-masalah yang lebih kecil dan spesifik:

1	Sub-masalah terkait Perilaku Pembeli (Siswa):
----------	--

2	Sub-masalah terkait Sistem Pelayanan Kantin:
3	Sub-masalah terkait Tata Letak Fasilitas/Fisik Kantin:

LANGKAH 3: PENGENALAN POLA DAN ABSTRAKSI

- **Pengenalan Pola:** Pernahkah kalian mengamati antrean panjang yang teratur di tempat lain (seperti di Loker Stasiun Kereta Api Cilacap atau di Minimarket)? Pola antrean atau sistem pelayanan apa yang mereka gunakan sehingga bisa tertib? Tuliskan analisis kelompokmu!
Jawab:
.....
.....
- **Abstraksi (Penyaringan Informasi):** Coretlah dengan spidol informasi di bawah ini yang menurut kelompokmu TIDAK RELEVAN / TIDAK PENTING dalam mencari solusi antrean kantin:

☐ Jumlah kasir pelayanan ☐ Warna cat dinding bangunan kantin ☐ Merk sepatu yang dipakai penjual
☐ Metode pembayaran jajan ☐ Jenis menu makanan yang dijual ☐ Luas area meja makan kantin

LANGKAH 4: MEMBIMBING PENYELIDIKAN (ANALISIS 5 WHYS)

Pilihlah salah satu sub-masalah dari Langkah 2 yang kelompokmu anggap paling krusial. Bongkar sub-masalah tersebut menggunakan rantai pertanyaan 5 Whys pada kolom di bawah ini hingga menemukan akar masalah terdalam!

Masalah yang Dipilih	:
WHY 1 (Mengapa?)	: Karena
WHY 2 (Mengapa?)	: Karena

WHY 3 (Mengapa?)	: Karena
WHY 4 (Mengapa?)	: Karena
WHY 5 (AKAR MASALAH)	: Ternyata akar penyebab utamanya adalah

LANGKAH 5: IDE SOLUSI LOGIS BERDASARKAN AKAR MASALAH

Berdasarkan akar masalah (Why 5) yang berhasil dikunci oleh kelompokmu, rumuskan satu ide solusi logis, kreatif, dan aplikatif (tanpa memerlukan komputer/biaya mahal) yang dapat segera diajukan ke sekolah!

Solusi Kelompok Kami:

.....
.....

LANGKAH 6: LEMBAR REFLEKSI MANDIRI (REGULASI DIRI INDIVIDU)

Bagian ini wajib diisi oleh masing-masing siswa secara mandiri (bukan diskusi kelompok):

1. Apakah selama kerja kelompok tadi saya benar-benar fokus mendengarkan ide teman saya atau cenderung memaksakan kehendak saya sendiri?

Jawab:

2. Apa pelajaran berharga yang saya dapatkan dari cara kerja metode 5 Whys dalam mengurai masalah rumit hari ini?

Jawab:

3. Bagaimana saya akan mempraktikkan cara berpikir detektif ini jika di masa depan saya menghadapi kesulitan belajar yang berat?

Jawab:

BAHAN BACAAN

MODUL MATERI: FONDASI COMPUTATIONAL THINKING & METODE 5 WHYS

Halo, para Detektif Masalah dan calon ahli teknologi masa depan!

Sebelum kita menyentuh *keyboard* untuk merancang kode program komputer atau membangun sistem Kecerdasan Artifisial (*Artificial Intelligence*), ada satu kemampuan fundamental yang wajib kalian kuasai: **Kemampuan Berpikir**.

Berpikir Komputasional (*Computational Thinking*) bukanlah tentang berpikir seperti mesin. Sebaliknya, ini adalah metode pemecahan masalah yang meniru cara seorang insinyur perangkat lunak dalam menstrukturkan dan menyelesaikan masalah rumit secara logis dan terencana.

Mari kita bedah satu per satu!

1. Masalah Sederhana (*Simple Problem*) vs Masalah Kompleks (*Complex Problem*)

Dalam kehidupan sehari-hari, baik di sekolah, di rumah, maupun di dunia industri nanti, kalian akan selalu berhadapan dengan masalah. Namun, mari kita pahami bahwa tidak semua masalah diciptakan setara.

A. Masalah Sederhana (*Simple Problem*)

Masalah sederhana adalah masalah yang sangat jelas ujung pangkalnya. Karakteristik utamanya meliputi:

- Memiliki **satu penyebab tunggal yang jelas**.
- Dapat langsung diselesaikan dengan **satu tindakan linier**.
- Solusinya bersifat pasti dan dapat diprediksi sejak awal.

Contoh: Lampu kamarmu mati karena sakelar belum ditekan, tali sepatumu lepas saat berjalan, atau ban sepedamu kekurangan angin. Solusinya? Tekan sakelar, ikat tali sepatu, dan pompa ban sepeda. Selesai!

B. Masalah Kompleks (*Complex Problem*)

Inilah tantangan yang sebenarnya di abad ke-21. Masalah kompleks adalah masalah yang ruwet, seperti benang kusut. Karakteristiknya meliputi:

- Penyebabnya saling berkaitan dan **berlapis-lapis**.
- Melibatkan **banyak variabel terikat**.
- Tidak memiliki satu solusi tunggal yang bisa langsung diterapkan.
- Jika kalian salah mengambil tindakan di awal, masalah tersebut justru bisa memburuk atau memicu masalah baru di tempat lain (dampak meluas).

Contoh: Masalah sampah plastik yang menumpuk di area kantin sekolah, kemacetan lalu lintas kota setiap jam berangkat kerja, atau menurunnya konsentrasi belajar siswa di kelas X secara massal.

Karakteristik	Masalah Sederhana	Masalah Kompleks
Penyebab	Tunggal dan langsung terlihat.	Berlapis, tersembunyi, banyak faktor.
Solusi	Satu tindakan langsung (Linier).	Butuh analisis berlapis dan bertahap.
Analogi	Mengganti ban bocor.	Mengatasi kemacetan lalu lintas kota.

2. Tiga Fondasi Utama Berpikir Komputasional (*Unplugged*)

Untuk mengurai masalah kompleks di atas tanpa menggunakan komputer (*unplugged*), kita butuh alat bantu berpikir. Tiga fondasi utamanya adalah:

□ A. Dekomposisi (*Decomposition*)

Dekomposisi adalah kemampuan emas untuk **memecah suatu masalah besar, rumit, dan menakutkan menjadi bagian-bagian yang jauh lebih kecil, spesifik, dan sederhana**. Tujuannya agar masalah tersebut lebih mudah dikelola dan diselesaikan satu per satu.

- **Analogi Logis:** Bayangkan kalian melihat sebuah robot mainan besar yang sangat rumit. Jika kalian langsung mencoba memahaminya keseluruhannya, otak akan kewalahan. Namun, melalui dekomposisi, proses melihat robot tersebut diubah menjadi susunan dari roda, baterai, kabel, *gear*, dan mur-mur kecil. Kalian memeriksa satu per satu bagian kecilnya secara runut.
- **Penerapan di SMK:** Kalian mendapat tugas membuat *website* sekolah. Jangan panik! Dekomposisi tugas tersebut menjadi: 1) Membuat desain tampilan, 2) Menulis isi teks, 3) Mencari foto kegiatan, 4) Membuat *database*.

□ B. Pengenalan Pola (*Pattern Recognition*)

Pengenalan pola adalah kemampuan otak kita untuk mencari **kesamaan, tren, atau karakteristik yang berulang di dalam sebuah masalah**, maupun membandingkannya dengan masalah lain yang pernah diselesaikan sebelumnya. Manusia dan komputer mengenali pola ini agar kita tidak perlu mencari solusi dari nol (dari awal) setiap kali menghadapi situasi baru.

- **Contoh Kasus:** Kalian menyadari sebuah pola bahwa motor selalu mogok *setiap kali* setelah menerjang genangan air hujan deras di wilayah Cilacap. Pola yang terbentuk: air selalu mengganggu sistem pengapian busi jika pelindungnya retak.
- **Penerapan di Koding:** Jika setiap kali kalian memasukkan angka desimal (koma) aplikasi kalian selalu *error*, kalian mengenali polanya: "Oh, program saya sepertinya hanya dirancang untuk membaca angka bulat (integer)."

□ C. Abstraksi (*Abstraction*)

Di dunia yang penuh dengan informasi, kemampuan mengabaikan itu sangat penting. Abstraksi adalah kemampuan untuk **menyaring informasi, hanya fokus pada hal-hal yang benar-benar esensial** (penting) untuk penyelesaian masalah, dan dengan berani **membuang atau mengabaikan detail-detail kecil yang tidak relevan**. Abstraksi mencegah otak kita dari *information overload* (kelebihan beban informasi).

- **Analogi Logis:** Saat kalian menggunakan *Google Maps* untuk mencari rute tercepat dari rumah ke SMK Muhammadiyah 1 Cilacap, kalian hanya peduli pada garis jalan raya dan persimpangan. Kalian sepenuhnya mengabaikan informasi tentang warna cat rumah penduduk, jumlah pohon kelapa, atau merk mobil yang lewat.
- **Penerapan di Basis Data:** Saat membuat kartu pelajar, informasi yang diabstraksi (diambil) hanyalah Nama, NISN, dan Kelas. Informasi tentang makanan favorit atau merk sepatu siswa diabaikan karena tidak relevan dengan tujuan kartu pelajar.

3. Metode 5 Whys (5 Mengapa): Senjata Pamungkas Menemukan Akar Masalah

Menemukan masalah itu mudah, tetapi menemukan **akar** dari masalah itu butuh teknik. Metode 5 Whys adalah teknik investigasi yang sangat ampuh dan sederhana, yang pertama kali diciptakan oleh **Sakichi Toyoda** (pendiri perusahaan Toyota).

Cara kerjanya adalah dengan mengajukan pertanyaan "**Mengapa?**" **secara berturut-turut minimal 5 kali** terhadap sebuah fenomena. Tujuannya adalah menembus lapisan permukaan masalah untuk menemukan **Akar Penyebab Utama (Root Cause)**.

PENTING: Jika kita hanya menyelesaikan masalah di permukaannya saja, masalah itu pasti akan muncul lagi di kemudian hari.

□♂ Studi Kasus Detektif 5 Whys: "Kasus Menurunnya Nilai Koding"

Mari kita praktikkan metode ini pada masalah nyata di sekolah!

- **Fenomena di Permukaan (Gejala):** Nilai praktik koding seorang siswa SMK merosot tajam. (*Reaksi biasa: Guru marah dan menuduh siswa malas belajar*).

Mari kita tahan amarah dan terapkan 5 Whys:

1. **Why 1 (Mengapa nilai praktiknya merosot?):** Karena dia terlambat mengumpulkan tugas *coding* mingguan.
2. **Why 2 (Mengapa dia terlambat mengumpulkan tugas?):** Karena aplikasi program koding di laptop sekolahnya sering *error* saat dia sedang bekerja.
3. **Why 3 (Mengapa aplikasinya sering error?):** Karena laptop yang dia gunakan memori penyimpanannya (RAM) penuh terisi *file* sampah.
4. **Why 4 (Mengapa memorinya penuh file sampah?):** Karena sistem laptop tersebut tidak pernah dilakukan pembersihan (*maintenance*) secara berkala.
5. **Why 5 (Mengapa tidak pernah dilakukan pembersihan berkala?):** □ **[AKAR MASALAH TEMUAN]:** Karena laboratorium komputer sekolah belum memiliki aturan tertulis atau *Standard Operating Procedure* (SOP) mengenai jadwal perawatan berkala untuk aset komputer