

Nama :

Kelas :

No. Absen :

Tanggal :

Kelompok :

TUJUAN PEMBELAJARAN

1. Siswa mampu menerapkan langkah-langkah berpikir komputasional dalam menyelesaikan masalah.
2. Siswa mampu menyusun algoritma dan flowchart untuk solusi masalah.
3. Siswa mampu membuat program sederhana berdasarkan algoritma yang telah dibuat.
4. Siswa mampu menguji dan memperbaiki (debugging) program dengan tepat.

APA ITU BERPIKIR KOMPUTASIONAL?

Berpikir komputasional adalah cara berpikir untuk menyelesaikan masalah secara logis, sistematis, dan efisien dengan langkah-langkah berikut:



DEKOMPOSISI
Memecah masalah menjadi bagian kecil



PENGENALAN POLA
Menemukan kesamaan atau pola



ABSTRAKSI
Memilih informasi penting, mengabaikan yang tidak relevan



ALGORITMA
Menyusun langkah penyelesaian masalah

PETUNJUK

- ☒ Kerjakan setiap misi secara berurutan.
- ☒ Diskusikan bersama anggota kelompok.
- ☒ Tulislah jawaban pada tempat yang sudah disediakan.
- ☒ Presentasikan hasil kerja kelompok di depan kelas.

Think Plan Code
Test Improve

MISI 1: DEKOMPOSISI MASALAH

Studi Kasus: Kantin sekolah ingin membuat sistem antrian agar pembeli dilayani secara tertib.

Tugas:

Tulislah dekomposisi masalah di bawah ini!

Masalah Utama

Bagaimana membuat sistem antrian di kantin sekolah?

Pecah Masalah Menjadi Bagian Kecil

1.
2.
3.
4.
5.

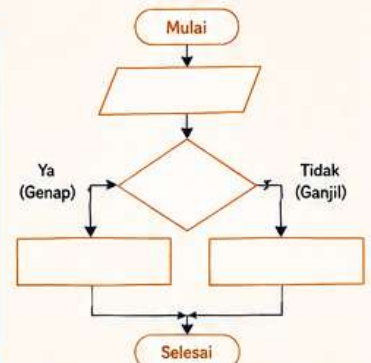
MISI 2: ALGORITMA & FLOWCHART

Kasus: Buat algoritma dan flowchart untuk menentukan apakah suatu bilangan adalah GENAP atau GANJIL.

Algoritma (Langkah-langkah)

1.
2.
3.
4.
5.
6.

Flowchart



MISI 3: DEBUGGING PROGRAM

Perhatikan kode program berikut! Program memiliki kesalahan. Temukan dan perbaiki kesalahannya!

Kode Program (Salah)

```

1 x = input("Masukkan bilangan: ")
2 if x % 2 = 0:
3     print("Bilangan Genap")
4 else
5     print('Bilangan Ganjil')
  
```

Kesalahan yang ditemukan:

1.
2.
3.

Kode Program (Perbaikan)

```

1
2
3
4
5
  
```

MISI 4: PROYEK MINI

Rancang program sederhana sesuai pilihan salah satu topik berikut. (1 kelompok memilih 1 topik)



A. KALKULATOR SEDERHANA

Program dapat melakukan operasi tambah, kurang, kali, bagi dua bilangan.



B. PENENTU NILAI HURUF

Program mengkonversi nilai angka ke huruf (A, B, C, D, E) sesuai kriteria.



C. ABSENSI SISWA

Program mencatat nama siswa yang hadir dan menampilkan jumlah siswa hadir.

Tuliskan rancangan singkat program kalian:



REFLEKSI

Setelah mengerjakan LKPD ini, tuliskan:

Hal baru yang saya pelajari:

Tantangan yang saya hadapi:

Manfaat materi ini untuk kehidupan saya:

PENILAIAN DIRI

Berilah tanda centang (✓) sesuai penilaianmu!

Aspek	Sangat Baik	Baik	Cukup	Perlu Perbaikan
Memahami konsep berpikir komputasional				
Menyusun algoritma dengan tepat				
Membuat program sederhana				
Kerja sama dalam kelompok				
Presentasi hasil kerja				

RUBRIK PENILAIAN GURU

Aspek	Skor Maks	Skor Diperoleh
Misi 1: Dekomposisi Masalah	20	
Misi 2: Algoritma & Flowchart	25	
Misi 3: Debugging Program	25	
Misi 4: Proyek Mini Program	20	
Presentasi	10	
Total	100	



Ingat! Programmer hebat bukan hanya pandai ngoding, tapi juga hebat memecahkan masalah!

Keep Thinking.
Keep Coding.
Keep Solving!



TUJUAN PEMBELAJARAN

- ✓ Siswa mampu menerapkan langkah-langkah berpikir komputasional.
- ✓ Siswa mampu menyusun algoritma dan flowchart untuk menyelesaikan masalah.
- ✓ Siswa mampu membuat program sederhana berdasarkan algoritma yang telah dibuat.
- ✓ Siswa mampu menguji dan memperbaiki program dengan tepat.

APA ITU BERPIKIR KOMPUTASIONAL?

Berpikir komputasional adalah cara berpikir untuk menyelesaikan masalah secara logis, sistematis, dan efisien dengan langkah-langkah berikut:



DEKOMPOSISI

Memecah masalah menjadi bagian-bagian kecil



PENGENALAN POLA

Menemukan kesamaan atau pola



ABSTRAKSI

Memilih informasi penting dan mengabaikan yang tidak relevan



ALGORITMA

Menyusun langkah penyelesaian masalah secara teratur

PETUNJUK

1. Bacalah setiap aktivitas dengan teliti.
2. Kerjakan secara berurutan dan diskusikan dengan anggota kelompok.
3. Tulislah jawaban pada tempat yang disediakan.
4. Presentasikan hasil kerja kelompok di depan kelas.

Ingat! Programmer hebat bukan hanya menulis kode, tapi juga pandai memecahkan masalah



AKTIVITAS 1 DEKOMPOSISI

Studi Kasus:

Sekolah akan membuat sistem peminjaman buku di perpustakaan secara digital.

Tugas:

Uraikan masalah tersebut menjadi bagian-bagian kecil!

Masalah Utama

Membuat sistem peminjaman buku digital

Pecah Masalah Menjadi Bagian Kecil

1. _____
2. _____
3. _____
4. _____
5. _____



AKTIVITAS 2 POLA & ABSTRAKSI

Dari kasus pada Aktivitas 1, temukan pola dari proses peminjaman buku di perpustakaan.

Kemudian tentukan informasi penting (abstraksi) yang diperlukan dalam sistem.

Pola yang Ditemukan

1. _____
2. _____
3. _____

Informasi Penting (Abstraksi)

- _____
- _____
- _____



Fokus pada hal penting, abaikan yang tidak relevan!



AKTIVITAS 3 ALGORITMA

Buatlah algoritma (langkah-langkah) untuk proses peminjaman buku.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____



AKTIVITAS 4 FLOWCHART

Gambarkan flowchart untuk algoritma pada Aktivitas 3!

KETERANGAN SIMBOL

- Terminator (Mulai/Selesai)
- Proses
- Input/Output
- Keputusan (Decision)
- Arah Alur



AKTIVITAS 5 DARI ALGORITMA KE KODE

Ubah algoritma pada Aktivitas 3 menjadi kode program sederhana (gunakan pseudocode atau bahasa Python). (Pilih salah satu)

Pseudocode

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____

Python (Contoh)

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____



AKTIVITAS 6 UJI & PERBAIKI

Perhatikan potongan kode berikut yang memiliki kesalahan. Temukan kesalahannya, jelaskan, dan perbaiki kodenya!

Kode Bermasalah

```
1 jumlah = input("Masukkan jumlah buku")
2 if jumlah >= 3
3     print("Boleh pinjam")
4 else
5     print("Tidak boleh pinjam")
```

a. Kesalahan yang ditemukan:

b. Jelaskan kesalahannya:

c. Perbaiki kode:

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____

REFLEKSI

Setelah mengerjakan LKPD ini, tuliskan pendapatmu!

1. Hal baru apa yang kamu pelajari?
2. Bagian mana yang paling menantang?
3. Bagaimana manfaat materi ini dalam kehidupan sehari-hari?



PENILAIAN DIRI

Berilah tanda centang (✓) sesuai dengan penilaianmu!

Aspek	Sangat Baik (4)	Baik (3)	Cukup (2)	Perlu Perbaikan (1)
Memahami konsep berpikir komputasional				
Menyusun algoritma dengan tepat				
Membuat flowchart dengan tepat				
Menulis kode program				
Bekerja sama dalam kelompok				
Presentasi hasil kerja				



PENILAIAN GURU

Aspek	Skor (0-20)	Catatan
Aktivitas 1 Dekomposisi		
Aktivitas 2 Pola & Abstraksi		
Aktivitas 3 Algoritma		
Aktivitas 4 Flowchart		
Aktivitas 5 Kode Program		
Aktivitas 6 Uji & Perbaiki		
Sikap & Kerja Sama		
TOTAL	100	





TUJUAN PEMBELAJARAN

- Memahami konsep berfikir komputasional dalam menyelesaikan masalah.
- Menyusun algoritma untuk menyelesaikan masalah secara sistematis.
- Membuat flowchart dari algoritma yang telah dirancang.
- Menguji dan memperbaiki algoritma sederhana.
- Menerapkan berfikir komputasional dalam situasi nyata.

APA ITU BERFIKIR KOMPUTASIONAL?

Berfikir komputasional adalah cara berpikir untuk menyelesaikan masalah secara logis, sistematis, dan efisien dengan langkah-langkah berikut:



PETUNJUK

- Bacalah setiap aktivitas dengan teliti.
- Kerjakan secara berurutan bersama anggota kelompok.
- Tuliskan jawaban pada tempat yang disediakan.
- Diskusikan dan presentasikan hasil kelompok di depan kelas.

Ingat! Programmer hebat bukan hanya menulis kode, tapi juga hebat dalam memecahkan masalah. ★



AKTIVITAS 1: DEKOMPOSISI MASALAH

Studi Kasus:

Sekolah ingin membuat sistem peminjaman buku di perpustakaan secara digital.

Tugas:

Uraikan masalah tersebut menjadi bagian-bagian kecil!

Masalah Utama

Membuat sistem peminjaman buku di perpustakaan secara digital

Pecah Masalah Menjadi Bagian Kecil

- _____
- _____
- _____
- _____
- _____
- _____



AKTIVITAS 2: PENGENALAN POLA

Dari daftar aktivitas peminjaman buku berikut, temukan pola atau kesamaan yang ada!

- Login anggota
- Cari buku
- Cek ketersediaan buku
- Pinjam buku
- Kembalikan buku
- Logout

Pola/Kesamaan yang ditemukan:

- _____
- _____
- _____



AKTIVITAS 3: ABSTRAKSI

Dari masalah sistem peminjaman buku, informasi manakah yang penting dan harus diperhatikan? Pilih yang relevan!

Daftar Informasi

- ☐ Nama anggota
- ☐ ID anggota
- ☐ Alamat rumah anggota
- ☐ Judul buku
- ☐ Penulis buku
- ☐ Tahun terbit buku
- ☐ Jumlah buku yang dipinjam
- ☐ Warna sampul buku
- ☐ Tanggal pinjam
- ☐ Tanggal kembali
- ☐ Denda keterlambatan

Informasi Penting (Abstraksi)

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____



AKTIVITAS 4: ALGORITMA

Buatlah algoritma (langkah-langkah) untuk proses peminjaman buku!

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____



AKTIVITAS 5: FLOWCHART

Gambarkan flowchart dari algoritma pada Aktivitas 4!



KETERANGAN SIMBOL

- Terminator (Mulai/Selesai)
- Proses
- Input/Output
- Decision (Keputusan)
- Flowline (Arah alur)

Tips: Pastikan alur dari Mulai sampai Selesai sudah terhubung!



AKTIVITAS 6: UJI DAN PERBAIKI

Perhatikan potongan kode berikut yang memiliki kesalahan. Temukan, jelaskan, dan perbaiki kesalahannya!

Kode Salah

```
1 jumlah = input("Masukkan jumlah buku")
2 if jumlah = 0 :
3     print("Tidak boleh 0")
4 else
5     harga = jumlah * 5000
6     total = jumlah + harga
7     print("Total =", total)
```

a. Kesalahan yang ditemukan:

b. Perbaikan yang dilakukan:

Perbaiki Kode

- _____
- _____
- _____
- _____
- _____
- _____
- _____



REFLEKSI

Jawablah pertanyaan berikut dengan jujur!

- Hal baru apa yang kamu pelajari hari ini?
- Bagian mana yang menurutmu paling menantang?
- Bagaimana berfikir komputasional dapat membantu kamu dalam kehidupan sehari-hari?

PENILAIAN DIRI

Berilah tanda centang (✓) sesuai pendapatmu!

Aspek Penilaian	Sangat Baik (4)	Baik (3)	Cukup (2)	Perlu Perbaikan (1)
Memahami konsep berfikir komputasional				
Mampu mendekomposisi masalah				
Menemukan pola yang relevan				
Membuat algoritma dengan tepat				
Membuat flowchart dengan benar				
Kerja sama dalam kelompok				
Presentasi hasil kerja				



RUBRIK PENILAIAN GURU

Aspek	Skor Maks	Skor Diperoleh	Catatan
Aktivitas 1 (Dekomposisi)	15		
Aktivitas 2 (Pengenalan Pola)	15		
Aktivitas 3 (Abstraksi)	15		
Aktivitas 4 (Algoritma)	20		
Aktivitas 5 (Flowchart)	20		
Aktivitas 6 (Uji & Perbaiki)	15		
TOTAL	100		



Berpikir komputasional adalah kunci untuk menjadi **problem solver** yang kreatif. Teruslah berlatih, karena setiap masalah pasti ada solusinya! ★





Pikir Logis • Pecahkan Masalah • Ciptakan Program

IDENTITAS SISWA

Nama : _____
Kelas : _____
No. Absen : _____
Tanggal : _____
Kelompok : _____



TUJUAN PEMBELAJARAN

- ✓ Siswa mampu menerapkan tahapan berfikir komputasional pada suatu masalah.
- ✓ Siswa mampu menyusun algoritma untuk menyelesaikan masalah secara sistematis.
- ✓ Siswa mampu membuat flowchart dari algoritma yang telah dibuat.
- ✓ Siswa mampu mengubah algoritma ke dalam bentuk kode program sederhana.



MATERI SINGKAT

Berfikir komputasional adalah cara berpikir untuk menyelesaikan masalah secara logis, sistematis, dan efisien dengan langkah-langkah berikut:



DEKOMPOSISI

Memecah masalah menjadi bagian-bagian kecil.



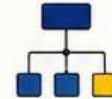
PENGENALAN POLA

Menemukan kesamaan atau pola dari masalah.



ABSTRAKSI

Memilih informasi penting, mengabaikan yang tidak relevan.



ALGORITMA

Menyusun langkah-langkah penyelesaian secara logis.



IMPLEMENTASI

Mewujudkan algoritma menjadi program dan menguji solusi.



MISI 1: SISTEM ABSENSI HARIAN

Sebuah sekolah ingin membuat program absensi sederhana. Program harus menginput jumlah siswa, lalu menginput status hadir (H) atau tidak hadir (T) untuk setiap siswa, kemudian menampilkan jumlah siswa hadir dan tidak hadir.

1 DEKOMPOSISI MASALAH

Uraikan masalah di atas menjadi bagian-bagian kecil!

- _____
- _____
- _____
- _____
- _____

2 PENGENALAN POLA

Pola apa yang kamu temukan dari masalah tersebut?

- _____
- _____
- _____

3 ABSTRAKSI

Informasi penting apa saja yang dibutuhkan?

- _____
- _____
- _____
- _____

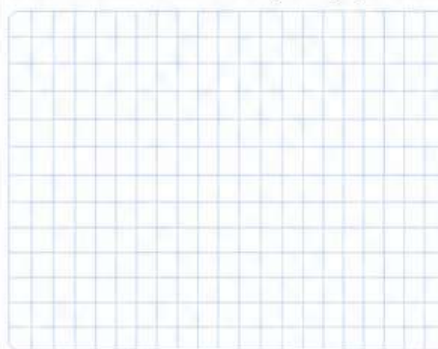
4 ALGORITMA (LANGKAH-LANGKAH)

Tuliskan algoritma langkah-langkah penyelesaian masalah!

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

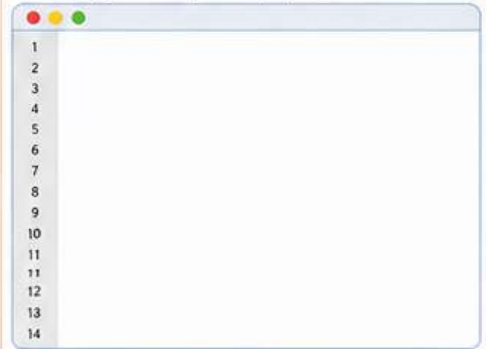
5 FLOWCHART

Gambarkan flowchart berdasarkan algoritma yang telah dibuat!



6 KODE PROGRAM (PYTHON)

Ubahlah algoritma menjadi kode program sederhana!



7 UJI COBA & HASIL

Ujicobalah programmu dengan contoh data berikut, lalu tuliskan hasilnya!

Contoh Data Uji

Jumlah siswa = 8
Status siswa = H, H, T, H, H, T, T, H

Hasil yang Diharapkan

Jumlah hadir = _____

Jumlah tidak hadir = _____

Hasil Programmu

Jumlah hadir = _____

Jumlah tidak hadir = _____

REFLEKSI

- Hal baru yang saya pelajari: _____
- Bagian yang menantang: _____
- Manfaat materi ini bagi saya: _____



Berfikir komputasional bukan hanya tentang kode, tapi tentang cara menemukan solusi terbaik!



TERUS BERLATIH, JADI PROBLEM SOLVER HEBAT!



ARDUINO

Tulis Kode, Kendalikan Dunia Nyata!

IDENTITAS SISWA

Nama : _____
 Kelas : _____
 No. Absen : _____
 Tanggal : _____
 Kelompok : _____



TUJUAN PEMBELAJARAN

- ✓ Siswa mampu mengenal lingkungan pemrograman Arduino IDE.
- ✓ Siswa mampu memahami struktur dasar program Bahasa C pada Arduino.
- ✓ Siswa mampu membuat program sederhana menggunakan Bahasa C Arduino.
- ✓ Siswa mampu meng-compile dan meng-upload program ke Arduino.



MATERI SINGKAT

Arduino menggunakan bahasa C/C++ yang sederhana dan mudah dipahami.

Struktur Dasar Program Arduino

```
void setup() {
  // kode di sini dijalankan satu kali saat awal
}

void loop() {
  // kode di sini dijalankan berulang-ulang
}
```

- void setup() : untuk inialisasi pin atau variabel.
- void loop() : untuk program utama yang diulang terus.
- Setiap perintah diakhiri dengan tanda titik koma (;).

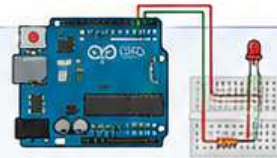
ELEMEN DASAR BAHASA C ARDUINO

- ☑ Komentar : // komentar satu baris
- ☑ Variabel : int, float, bool, char
- ☑ Operator : +, -, *, /, %, ==, !=, >, <, &&, ||
- ☑ Fungsi I/O : digitalWrite(), digitalRead(), analogRead(), analogWrite()
- ☑ Konstanta : HIGH, LOW, true, false
- ☑ Tipe Data Umum : int, float, bool, char
- ☑ Contoh Output : Serial.begin(9600);
Serial.println("Halo");



MISI 1: LED PERTAMAKU BERKEDIP

Pada misi ini, kamu akan membuat program sederhana untuk menyalakan LED yang terhubung pada pin 13 Arduino secara berkedip. LED akan menyala selama 1 detik, kemudian mati selama 1 detik, dan berulang terus.



Rangkaian:

- LED (+) → Resistor 220Ω → Pin 13
- LED (-) → GND

1 DEKOMPOSISI MASALAH

Uraikan masalah di atas menjadi bagian-bagian kecil!

- _____
- _____
- _____
- _____
- _____



2 PENGENALAN POLA

Pola atau kesamaan apa yang kamu temukan dalam masalah tersebut?

- _____
- _____
- _____



3 ABSTRAKSI

Informasi penting apa saja yang perlu diperhatikan?

- _____
- _____
- _____
- _____



4 ALGORITMA (LANGKAH-LANGKAH)

Tuliskan algoritma langkah-langkah untuk mengedipkan LED!

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____



5 KODE PROGRAM

Lengkapilah kode program berikut agar LED pada pin 13 berkedip setiap 1 detik!

```
1 void setup() {
2   pinMode(13, OUTPUT);
3 }
4
5 void loop() {
6   digitalWrite(13, HIGH); // LED ON
7   _____ (1000); // tunggu 1 detik
8   digitalWrite(13, LOW); // LED OFF
9   _____ (1000); // tunggu 1 detik
10 }
11
12
13
14
```

6 HASIL & VERIFIKASI

Upload program ke Arduino, amati LED, kemudian isi tabel hasil pengamatan!

Percobaan ke-	LED Menyala?	LED Mati?	Sesuai Program?	Catatan
1	☐	☐	☐	
2	☐	☐	☐	
3	☐	☐	☐	



7 REFLEKSI

Jawablah pertanyaan berikut!

1. Apa yang kamu pelajari tentang struktur dasar program Arduino? _____
2. Mengapa fungsi setup() hanya dijalankan sekali? _____
3. Apa manfaat menggunakan delay() dalam program? _____
4. Bagaimana perasaanmu setelah berhasil membuat LED berkedip? _____



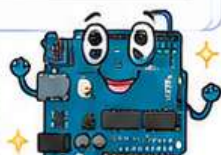
TIPS SUKSES

Compile (✓) → Upload (→) → Amati (👁) → Evaluasi (📝) → Ulangi!



Terus berlatih dan bereksperimen!

Setiap baris kode adalah langkah menuju inovasi.



Nama : _____
 Kelas : _____
 No. Absen : _____
 Tanggal : _____
 Kelompok : _____



TUJUAN PEMBELAJARAN

- ✓ Siswa mampu menerapkan tahapan berfikir komputasional dalam menyelesaikan masalah.
- ✓ Siswa mampu menyusun algoritma untuk menyelesaikan masalah secara sistematis.
- ✓ Siswa mampu membuat flowchart dari algoritma yang telah dibuat.
- ✓ Siswa mampu menguji dan memperbaiki algoritma sederhana.

MATERI SINGKAT

Berfikir komputasional adalah cara berfikir untuk menyelesaikan masalah secara logis, sistematis, dan efisien. Tahapannya meliputi:



DEKOMPOSISI

Memecah masalah menjadi bagian-bagian kecil.



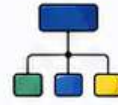
PENGENALAN POLA

Menemukan kesamaan atau pola dari masalah.



ABSTRAKSI

Memilih informasi penting, mengabaikan yang tidak relevan.



ALGORITMA

Menyusun langkah-langkah penyelesaian secara logis.



IMPLEMENTASI

Mewujudkan algoritma menjadi program dan menguji solusi.

MISI 1



MENENTUKAN NILAI AKHIR DAN PREDIKAT SISWA

Sebuah sekolah ingin membuat program untuk menentukan nilai akhir dan predikat siswa berdasarkan nilai Tugas (T), Ulangan (U), dan Ujian Akhir (UA).

Nilai akhir dihitung dengan rumus:

$$\text{Nilai Akhir} = (30\% \times T) + (30\% \times U) + (40\% \times UA)$$

Predikat nilai berdasarkan ketentuan berikut:

Nilai Akhir	Predikat	Keterangan
≥ 85	A	Sangat Baik
70 – 84	B	Baik
55 – 69	C	Cukup
40 – 54	D	Kurang
< 40	E	Sangat Kurang

TUGAS

Gunakan tahapan berfikir komputasional untuk membuat solusi dari permasalahan di samping!



1 DEKOMPOSISI MASALAH

Uraikan masalah di atas menjadi bagian-bagian kecil!

- _____
- _____
- _____
- _____
- _____

2 PENGENALAN POLA

Apa pola atau kesamaan yang kamu temukan dari masalah tersebut?

- _____
- _____
- _____

3 ABSTRAKSI

Informasi penting apa saja yang perlu diperhatikan? (Tuliskan yang relevan saja!)

- _____
- _____
- _____

4 ALGORITMA (LANGKAH-LANGKAH)

Tuliskan algoritma langkah-langkah penyelesaian masalah secara berurutan!

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____



5 FLOWCHART

Gambarkan flowchart dari algoritma yang telah kamu buat!



6 KODE PROGRAM (PSEUDOCODE)

Ubahlah algoritma menjadi pseudocode atau kode program sederhana!



7 UJI COBA & HASIL

Gunakan data contoh berikut, lalu hitung nilai akhir dan predikatnya!

No.	Tugas (T)	Ulangan (U)	Ujian Akhir (UA)	Nilai Akhir	Predikat
1.	80	70	90	_____	_____
2.	60	65	70	_____	_____
3.	40	50	55	_____	_____



8 REFLEKSI

a. Apa kesulitan yang kamu hadapi saat menyelesaikan tugas ini?

b. Bagian mana yang paling membantu kamu dalam menyelesaikan masalah?

c. Bagaimana berfikir komputasional dapat membantu dalam kehidupan sehari-hari?



Berfikir komputasional adalah kunci untuk memecahkan masalah secara cerdas!

Terus berlatih dan jadilah problem solver yang hebat! 🌟



Nama :
Kelas :
No. Absen :
Tanggal :
Guru :

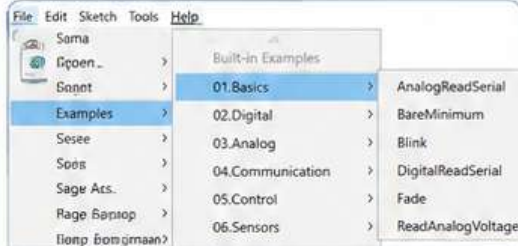


TUJUAN PEMBELAJARAN

- ✓ Siswa mampu menginstal Arduino IDE pada komputer dengan benar.
- ✓ Siswa mampu mengidentifikasi komponen utama pada tampilan Arduino IDE.
- ✓ Siswa mampu membuka menu Examples sebagai langkah awal mengenal pemrograman Arduino.
- ✓ Siswa mampu mendokumentasikan proses instalasi dalam bentuk langkah-langkah detail.

MATERI SINGKAT

Arduino IDE (*Integrated Development Environment*) adalah perangkat lunak resmi dari Arduino yang digunakan untuk menulis, mengompilasi, dan mengunggah program ke board Arduino. Setelah terinstal, kita dapat mencoba berbagai contoh program melalui menu **Examples**.



Menu Examples pada Arduino IDE

Komponen Tampilan IDE

1. Toolbar : Verifikasi, Upload, New, Open, Save, Serial Monitor
2. Editor Program : Tempat menulis kode
3. Message Area : Menampilkan proses kompilasi / upload
4. Console : Menampilkan pesan detail (error/success)
5. Board & Port : Pilihan perangkat yang digunakan

AKTIVITAS INDIVIDU



Ayo Lakukan!

No Aktivitas: SK-K12-02

Lakukan instalasi IDE Arduino pada komputer kalian. Kemudian tuliskan langkah-langkah detailnya pada jurnal kalian. Jika kalian tidak memiliki komputer, kalian dapat mencari dan menyaksikan videonya melalui internet atau melalui video yang disiapkan guru kalian.



Pindai QR Code untuk video tutorial instalasi Arduino IDE!

1. LANGKAH-LANGKAH INSTALASI ARDUINO IDE

Tuliskan langkah-langkah instalasi Arduino IDE di bawah ini secara urut dan detail!

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

Catatan Penting:

Pastikan komputer kalian terhubung ke internet saat mengunduh Arduino IDE. Pilih versi yang sesuai dengan sistem operasi (Windows, macOS, atau Linux).



2. BUKA MENU EXAMPLES

Setelah Arduino IDE berhasil dibuka, buka menu **Examples**. Isi tabel berikut dengan contoh-contoh yang kamu temukan pada menu **01.Basics**!



No.	Nama Contoh Program	Fungsi Singkat
1.		
2.		
3.		
4.		
5.		
6.		

3. HASIL DAN DOKUMENTASI

- Apakah Arduino IDE berhasil terinstal? ☐ Ya ☐ Tidak
- Versi Arduino IDE yang terinstal:
- Tampilan utama Arduino IDE (screenshot):



4. REFLEKSI

Jawablah pertanyaan berikut dengan jujur!

1. Apa kesulitan yang kamu temui saat instalasi Arduino IDE?
2. Bagaimana cara kamu mengatasinya?
3. Apa manfaat mempelajari Arduino IDE bagi kamu?



5. CHECKLIST PENILAIAN DIRI

Berikan tanda centang (✓) sesuai dengan pencapaianmu!



Aspek yang Dinilai	Sudah (✓)	Belum
Instalasi Arduino IDE berhasil		
Dapat membuka Arduino IDE		
Dapat membuka menu Examples		
Menuliskan langkah instalasi dengan lengkap		
Memahami fungsi tampilan utama IDE		



Terus belajar dan berlatih! Setiap langkah kecilmu hari ini adalah kemajuan besar untuk masa depanmu!



Nama : _____
 Kelas : _____
 No. Absen : _____
 Tanggal : _____
 Guru : _____



TUJUAN PEMBELAJARAN

- ✓ Siswa mampu menjelaskan struktur dasar program Arduino: `setup()` dan `loop()`.
- ✓ Siswa mampu mengidentifikasi bagian-bagian struktur program Arduino.
- ✓ Siswa mampu membuat program sederhana dengan struktur dasar yang benar.
- ✓ Siswa mampu menganalisis fungsi kode berdasarkan struktur program.

MATERI SINGKAT

Setiap program Arduino selalu memiliki dua bagian utama, yaitu `setup()` dan `loop()`.

```
1 void setup() {
2   // kode di sini dijalankan satu kali saat pertama kali
3   // board Arduino menyala atau di-reset
4 }
5 void loop() {
6   // kode di sini dijalankan berulang-ulang
7   // selama board Arduino menyala
8 }
```

Keterangan

Bagian	Fungsi
<code>void setup()</code>	Digunakan untuk inisialisasi. Hanya dijalankan sekali saat program dimulai atau setelah reset.
<code>void loop()</code>	Digunakan untuk program utama. Dijalankan terus-menerus selama Arduino aktif.

AKTIVITAS INDIVIDU



Ayo Lakukan!

No Aktivitas: SK-K12-03



Tips Belajar!

Perhatikan urutan kode dan tanda kurung kurawal { } yang menandai awal dan akhir blok program.

Pahami struktur dasar program Arduino, kemudian kerjakan tugas-tugas berikut secara individu. Tuliskan jawabanmu pada tempat yang telah disediakan.

1 MENGENAL STRUKTUR DASAR

Lengkapi keterangan fungsi dari bagian-bagian struktur program berikut!

```
1 void setup() {
2   _____
3 }
4 void loop() {
5   _____
6 }
7
```

Keterangan Fungsi

- a. `void setup()` : _____
- b. `void loop()` : _____

2 IDENTIFIKASI BAGIAN PROGRAM

Perhatikan potongan program berikut! Tuliskan bagian kode yang termasuk `setup()` dan `loop()`.

```
1 void setup() {
2   pinMode(13, OUTPUT);
3   Serial.begin(9600);
4 }
5
6 void loop() {
7   digitalWrite(13, HIGH);
8   delay(1000);
9   digitalWrite(13, LOW);
10  delay(1000);
11 }
```

a. Bagian `setup()`

b. Bagian `loop()`

3 ANALISIS KODE

Jelaskan fungsi dari potongan kode program berikut!

```
1 void setup() {
2   pinMode(13, OUTPUT);
3 }
4 void loop() {
5   digitalWrite(13, HIGH);
6   delay(1000);
7 }
```

Penjelasan:

4 MENYUSUN STRUKTUR PROGRAM

Susun potongan kode di bawah ini menjadi struktur program Arduino yang benar!

```
digitalWrite(13, LOW);
void loop() {
  delay(1000);
  digitalWrite(13, HIGH);
}
void setup() {
  }
pinMode(13, OUTPUT);
delay(1000);
```



1
2
3
4
5
6
7
8
9
10

5 MEMBUAT PROGRAM SEDERHANA

Buatlah program Arduino untuk menyalakan LED pada pin 13 setiap 0,5 detik!

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

7 REFLEKSI

Jawablah pertanyaan berikut dengan jujur!



1. Apa hal baru yang kamu pelajari hari ini? _____
2. Bagaimana pemahamanmu tentang struktur dasar program Arduino? _____
3. Apa manfaat memahami struktur dasar program Arduino? _____

CHECKLIST DIRI

Berikan tanda centang (✓) pada aspek yang sudah kamu kuasai!

- ☐ Saya dapat menjelaskan fungsi `setup()` dan `loop()`.
- ☐ Saya dapat mengidentifikasi bagian `setup()` dan `loop()` dalam kode.
- ☐ Saya dapat membuat program sederhana dengan struktur yang benar.

