

LEMBAR ASESMEN PESERTA DIDIK (LAPD)

Pengembangan Aplikasi Berbasis C++



Instansi : SMA Negeri 1 Bandung



Mata Pelajaran : Informatika



Kelas : XI / Fase F



Alokasi Waktu : 4 x 40 menit (1 Pertemuan)



Model Pembelajaran : *Project-Based Learning*



Tools : OnlineGDB · Padlet · Canva · LMS Elmu



IDENTITAS KELOMPOK

Nama Kelompok :

Anggota 1 (Programmer) :

Anggota 2 (Analisis) :

Anggota 3 (Presenter) :

Anggota 4 (Debugger) :

Anggota 5 (Dokumentator) :

Tanggal :



PERTANYAAN PEMANTIK

“Bagaimana teknologi di bidang yang kamu minati berdampak pada masyarakat, dan solusi komputasional apa yang dapat kamu rancang untuk merespons masalah tersebut?”

Diskusikan dengan kelompokmu selama 3 menit, lalu catat respons awal di bawah ini.



Bidang teknologi yang dipilih kelompok dan alasannya:

.....
.....
.....



Satu masalah nyata di bidang tersebut yang ingin diselesaikan:

.....
.....
.....

FASE-FASE PjBL



C++



1 Identifikasi Masalah & Riset (20 Menit)



Lakukan riset singkat melalui internet/Padlet untuk memahami masalah yang dipilih kelompokmu.

Tuliskan 3 fakta/data tentang masalah yang kamu temukan dari sumber terpercaya (sertakan sumber!):

1.

Sumber: _____

2.

Sumber: _____

3.

Sumber: _____

Dekomposisi masalah, pecah masalah besar menjadi sub-masalah yang bisa diselesaikan program:

- _____
- _____
- _____
- _____

Checkpoint Formatif 1 - Guru memvalidasi sebelum lanjut ke Fase 2

- Masalah sudah spesifik dan dapat dikomputasikan
- Sumber informasi valid dan tercantum
- Sub-masalah logis dan terhubung ke masalah utama

2 Perancangan Algoritma (Pseudocode) (20 Menit)

Sebelum menulis kode, rancang alur logika program dengan pseudocode. Pilih level tugas sesuai kemampuan anggotamu.

Level 1: Eksplorasi

Level 2: Pengembangan

Level 3: Tantangan

Lengkapi pseudocode berikut dengan mengisi bagian yang kosong:

MULAI

Tampilkan "Selamat Datang"

Input _____

Jika _____ maka

Tampilkan _____

Jika tidak

Tampilkan _____

SELESAI

Buat pseudocode lengkap untuk program yang:

- Menerima input dari pengguna
- Memproses data dengan minimal 1 kondisi (if/else)
- Menggunakan minimal 1 perulangan (for/while)
- Menampilkan output yang bermakna

Buat pseudocode untuk program dengan:

- Fungsi/prosedur terpisah
- Penanganan input tidak valid
- Minimal 2 fitur berbeda

Rancang test case untuk menguji setiap kondisi



Tuliskan pseudocode kelompokmu di sini (tanda level yang dipilih):

// Tuliskan pseudocode di sini...

// Level yang dipilih: Level 1 Level 2 Level 3

Checkpoint Formatif 2

- Pseudocode runtut dan dapat dipahami
- Logika sesuai dengan masalah yang diidentifikasi di Fase 1
- Level tugas sesuai kemampuan anggota

3 Implementasi Program (C++ via OnlineGDB) (35 Menit)

Terjemahkan pseudocode menjadi kode C++. Gunakan pola: Masalah → Pseudocode → Kode.



Salin/screenshot kode C++ yang sudah berjalan (tempel link OnlineGDB jika ada):

// Paste kode C++ atau link OnlineGDB:
// [https://www.onlinegdb.com/...](https://www.onlinegdb.com/)



Ingat: Error bukan kegagalan error adalah informasi. Setiap error yang berhasil diperbaiki merupakan kemajuan nyata dalam pemrograman.

Catat error yang ditemukan dan cara mengatasinya (min. 1 error):

| Pesan Error / Bug | Penyebab (Analisis) | Cara Memperbaiki |
|-------------------|---------------------|------------------|
| | | |
| | | |
| | | |
| | | |

Checkpoint Formatif 3

- Program dapat dijalankan (compile + run tanpa error fatal)
- Output sesuai dengan tujuan program
- Minimal 1 error tercatat dan dijelaskan

4 Pengujian (Testing & Debugging)



Uji program dengan berbagai skenario input untuk memastikan ketepatan dan ketahanan program.

Level 1

Uji dengan 2 input berbeda:

- Input normal (yang diharapkan)
- Input angka negatif atau nol

Level 2

Uji dengan 4 skenario:

- Input valid batas bawah
- Input valid batas atas
- Input tidak valid (huruf/kosong)
- Input di luar rentang yang diharapkan

Level 3: Tantangan

Buat tabel test case terstruktur:

- Input, Expected Output, Actual Output, Status (Pass/Fail)
- Min 6 test case mencakup semua kondisi program

Tabel hasil pengujian:

| No | Skenario/Input | Output yang diharapkan | Output Aktual | Status |
|----|----------------|------------------------|---------------|-----------|
| 1 | | | | Pass/Fail |
| 2 | | | | Pass/Fail |
| 3 | | | | Pass/Fail |
| 4 | | | | Pass/Fail |

5 Presentasi & Komunikasi Hasil



Buat slide presentasi (Canva) dan presentasikan hasil proyek kepada kelas.

Gunakan struktur berikut:



Masalah: Apa masalah yang dipilih? Mengapa penting? (data/fakta)



Solusi: Bagaimana program kamu menyelesaikan masalah tersebut?



Demonstrasi: Tunjukkan program berjalan live atau screenshot output



Kendala & Pembelajaran: Error apa yang ditemukan? Apa yang dipelajari?



Pengembangan Selanjutnya: Jika ada lebih banyak waktu, apa yang akan ditambahkan?






Link slide Canva / LMS submission:

6

LEMBAR KONTRIBUSI INDIVIDU (DIISI OLEH KETUA / GURU)

Skala: 4 = Sangat Aktif · 3 = Aktif · 2 = Cukup · 1 = Kurang Aktif



| Nama/Peran | Fase 1 Riset | Fase 2 Algoritma | Fase 3 Koding | Fase 4 Testing | Fase 5 Presentasi | Total | Catatan |
|--|--------------|------------------|---------------|----------------|-------------------|-------|---------|
|  Programmer | | | | | | | _____ |
|  Debugger | | | | | | | _____ |
|  Analis | | | | | | | _____ |
|  Dokumentator | | | | | | | _____ |
|  Presenter | | | | | | | _____ |

7 RUBRIK PENILAIAN



| Aspek | 4 Sangat Baik | 3 Baik | 2 Cukup | 1 Perlu Bimbingan |
|--|--|--|--|--|
| Identifikasi Masalah (Fase 1)  | Masalah spesifik, didukung 3+ fakta dari sumber valid, dekomposisi logis dan lengkap | Masalah jelas, 2 fakta dari sumber valid, dekomposisi cukup logis | Masalah kurang spesifik, sumber tidak tercantum, dekomposisi minimal | Masalah tidak relevan atau tidak ada dekomposisi |
| Kualitas Algoritma (Fase 2)  | Pseudocode runtut, logis, sesuai level, mencerminkan solusi masalah dengan baik | Pseudocode runtut dan sesuai level dengan sedikit ketidakkonsistenan | Pseudocode ada tapi ada lompatan logika atau tidak sesuai level | Pseudocode tidak ada atau tidak dapat dipahami |
| Implementasi Kode (Fase 3)  | Kode berjalan, output tepat, error tercatat dan dianalisis dengan baik | Kode berjalan, output sebagian tepat, minimal 1 error tercatat | Kode ada namun ada error yang belum teratasi, analisis error minimal | Kode tidak ada atau tidak dapat dijalankan |
| Pengujian (Fase 4)  | ≥4 skenario uji terstruktur, mencakup edge case, analisis pass/fail jelas | 3 skenario uji relevan dengan analisis cukup | 2 skenario uji tapi kurang bervariasi | Hanya 1 skenario atau tidak ada pengujian |
| Refleksi Individu  | Refleksi mendalam, spesifik, menunjukkan kesadaran diri dan growth mindset | Refleksi jelas dan relevan dengan pengalaman belajar | Refleksi ada tapi dangkal atau generik | Refleksi tidak diisi atau tidak relevan |
| Kontribusi Individu  | Aktif di semua fase, peran dijalankan penuh, mendukung anggota lain | Aktif di sebagian besar fase, peran cukup terpenuhi | Kontribusi tidak merata, lebih banyak mengikuti | Kontribusi sangat minim di semua fase |



Ingat: Penilaian ini bertujuan untuk mendukung proses belajar dan perbaikan berkelanjutan.
Teruslah belajar, berkolaborasi, dan berkarya!

