



## Game Tebak Angka Multipemain dengan Penilaian Skor Otomatis Menggunakan Bahasa C++

**Sheila Caroline Putri Wardana<sup>1</sup>, Shafra Syazwina Jafni Nur Sabrina<sup>2</sup>**

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro PSDKU Kediri, Kediri, Indonesia

---

### Artikel Info

#### Kata kunci:

Game edukasi  
C++  
Sorting  
Searching  
Pointer

---

### ABSTRAK

Permainan "Tebak Angka" merupakan salah satu media pembelajaran logika dan algoritma dalam pemrograman. Penelitian ini bertujuan mengembangkan program game edukatif berbasis C++ dengan fitur multipemain dan sistem penilaian otomatis berbasis jumlah percobaan. Dalam pengembangannya, digunakan struktur data struct, array, pointer, serta fungsi seperti sorting, searching, dan rekursif. Metode bubble sort dipakai untuk pengurutan skor, sedangkan pencarian nama pemain menggunakan linear search. Hasil program diuji dengan beberapa skenario dan menunjukkan kinerja yang sesuai, serta dapat menjadi media pembelajaran interaktif untuk memahami logika dasar pemrograman. Program juga memberikan pengalaman bermain yang menyenangkan serta memperkuat pemahaman logika pemrograman.

---

#### Penulis Korespondensi :

Sheila Caroline Putri Wardana, Shafra Syazwina Jafni Nur Sabrina  
Program Studi Teknik Informatika  
Fakultas Ilmu Komputer  
Universitas Dian Nuswantoro, Kediri

---

### 1. PENDAHULUAN

Bahasa pemrograman C++ merupakan salah satu bahasa pemrograman tingkat tinggi yang banyak digunakan dalam dunia pendidikan untuk memperkenalkan konsep dasar logika dan algoritma. Pembelajaran pemrograman dasar kerap kali dianggap sulit oleh sebagian mahasiswa, terutama ketika harus memahami konsep abstrak seperti pointer, struktur data, dan fungsi rekursif [1], [2], [3].

Game edukatif adalah salah satu metode alternatif pembelajaran yang menyenangkan dan interaktif [4]. Dengan memadukan unsur hiburan dan edukasi, game dapat membantu meningkatkan minat belajar serta memperkuat pemahaman konsep melalui simulasi nyata [5], [6]. Salah satu contoh permainan sederhana yang efektif adalah game "Tebak Angka", di mana pemain harus menebak angka secara acak yang ditentukan oleh program [7], [8].

Penelitian ini bertujuan untuk membangun aplikasi permainan edukatif "Tebak Angka" berbasis C++ dengan fitur multipemain dan sistem penilaian otomatis [9]. Fokus utama dari implementasi ini adalah penggunaan konsep pemrograman dasar seperti struct, array, pointer, fungsi prosedural dan rekursif, serta teknik sorting dan searching [10].

## 1.1 Tinjauan Pustaka

Tinjauan pustaka berfungsi untuk memberikan landasan teori serta studi-studi sebelumnya yang relevan dengan penelitian atau pengembangan yang dilakukan. Pada penelitian ini, terdapat beberapa konsep utama yang dijadikan acuan:

### a. Game Edukatif dalam Pembelajaran

Game edukatif merupakan sarana pembelajaran berbasis digital yang menggabungkan unsur hiburan dan pendidikan untuk meningkatkan motivasi belajar. Game semacam ini terbukti mampu meningkatkan partisipasi siswa dalam memahami materi yang dianggap sulit. Dalam konteks pembelajaran pemrograman, pendekatan berbasis game mampu memberikan pengalaman praktik nyata yang menyenangkan [11], [12].

### b. Bahasa Pemrograman C++

C++ adalah bahasa pemrograman tingkat menengah yang mendukung paradigma prosedural maupun berorientasi objek. C++ banyak digunakan dalam pembelajaran dasar karena memiliki fitur lengkap seperti struktur data, pointer, fungsi, hingga manipulasi memori secara manual. Bahasa pemrograman C++ dirancang untuk memberikan efisiensi tinggi dalam pemrosesan dan fleksibilitas dalam desain perangkat lunak [13], [14], [15].

### c. Struktur Data: Struct, Array, Pointer

**Struct** merupakan tipe data bentukan yang memungkinkan penyimpanan beberapa jenis data dalam satu entitas, sangat cocok untuk representasi data pemain seperti nama dan skor.

**Array** digunakan sebagai kontainer dari data bertipe struct, memungkinkan penyimpanan data banyak pemain secara berurutan.

**Pointer** memberikan akses langsung ke memori dan efisiensi dalam pengolahan data antar fungsi, seperti yang digunakan pada fungsi mainGame().

### d. Algoritma Bubble Sort dan Linear Search

**Bubble Sort** adalah metode pengurutan data sederhana namun efektif untuk dataset kecil. Penggunaannya sesuai untuk mengurutkan pemain berdasarkan skor atau jumlah percobaan.

**Linear Search** digunakan untuk pencarian nama karena cocok untuk dataset kecil dan implementasinya mudah dipahami oleh pemula.

### e. Rekursi dalam Penghitungan Skor

Konsep rekursi adalah fungsi yang memanggil dirinya sendiri. Penggunaan rekursi dalam program ini dimaksudkan agar pengguna memahami penerapan fungsi rekursif dalam perhitungan, seperti penilaian otomatis berdasarkan jumlah percobaan.

## 2. METODE

### 2.1 Struct ADT

**Struct** digunakan untuk membuat tipe data abstrak yang merepresentasikan pemain. Dalam program ini, tipe struct Pemain menyimpan informasi dasar yaitu nama dan jumlah percobaan yang dilakukan pemain dalam menebak angka.

```

8  struct Pemain {
9      string nama;
10     int percobaan;
11     int skor;
12 };
13

```

Gambar 1. Penggunaan Struct

Digunakan struct "Pemain" untuk menyimpan nama dan jumlah percobaan. Array dari struct digunakan sebagai kontainer data untuk seluruh pemain.

## 2.2 Pointer

**Pointer** digunakan untuk memanipulasi data secara langsung dari fungsi-fungsi prosedural. Hal ini dilakukan untuk menghindari duplikasi data dan memberikan kontrol penuh terhadap nilai yang diubah dalam array Pemain.

```

20 void mainkanPemain(Pemain* p, int angka) {
50
51 for (int i = 0; i < n; i++) {
52     cout << "\nMasukkan nama pemain " << i+1 << ": ";
53     cin >> pemain[i].nama;
54     int angka = rand() % 100 + 1;
55     mainkanPemain(&pemain[i], angka); // Pakai pointer
56 }

```

Gambar 2. Penggunaan Pointer

Pointer dipakai dalam fungsi prosedural seperti mainGame(Pemain\* p) untuk memungkinkan pengubahan data secara langsung.

## 2.3. Fungsi Prosedural dan Rekursif

Fungsi prosedural digunakan untuk membagi program menjadi bagian-bagian modular seperti menjalankan permainan, mengurutkan data, mencari pemain, dan menghitung skor. Hal ini mempermudah pemeliharaan dan pemahaman alur program. Salah satu contohnya adalah fungsi mainGame() yang menangani proses permainan untuk tiap pemain. Sementara itu, hitungSkor() menggunakan logika rekursif sederhana untuk menghitung skor berdasarkan jumlah percobaan, sekaligus memperkenalkan konsep rekursi secara praktis dalam konteks pemrograman.

```

13
14 int hitungSkor(int percobaan) {
15     if (percobaan <= 1) return 100;
16     int sisa = hitungSkor(percobaan - 1) - 5;
17     return (sisa < 0) ? 0 : sisa;
18 }

```

Gambar 3. Hitung Skor

```

21 void mainkanPemain(Pemain* p, int angka) {
22
23     // Sorting berdasarkan skor (bubble sort)
24     void urutkanSkor(Pemain data[], int n) {
25
26         for (int i = 0; i < n-1; i++) {
27             for (int j = 0; j < n-i-1; j++) {
28
29                 if (data[j].skor < data[j+1].skor) {
30                     Pemain temp = data[j];
31                     data[j] = data[j+1];
32                     data[j+1] = temp;
33
34                 }
35             }
36         }
37     }
38 }
39

```

Gambar 4. Urutkan Skor

Program terdiri dari beberapa fungsi seperti mainGame(), sorting(), cariPemain(), dan hitungSkor(int percobaan). Fungsi hitungSkor menerapkan logika penilaian rekursif.

#### 2.4. Sorting dan Searching

**Bubble Sort:** Digunakan untuk mengurutkan pemain berdasarkan jumlah percobaan dari yang terbaik (sedikit percobaan) ke terburuk. Bubble Sort dipilih karena kesederhanaannya dan cukup efisien untuk jumlah data yang relatif kecil .

**Linear Search:** Untuk pencarian nama pemain. Metode ini cukup karena pencarian dilakukan sekaligus dan pada dataset kecil.

```

64 int cariPemain(Pemain data[], int n, string kunci) {
65
66     for (int i = 0; i < n; i++) {
67
68         if (data[i].nama == kunci) return i;
69     }
70 }

```

Gambar 5. Cari Pemain

Data pemain diurutkan dengan metode Bubble Sort berdasarkan jumlah percobaan. Untuk pencarian nama, digunakan metode Linear Search.

#### 2.5. Implementasi Alur Program

Input jumlah pemain dan nama

```

Masukkan jumlah pemain (maks 10): 2
Masukkan nama pemain 1: carol

```

Gambar 6. Input Jumlah Pemain

Pada tahap awal, program meminta input dari pengguna mengenai jumlah pemain yang akan ikut bermain. Setelah itu, nama dari masing-masing pemain dimasukkan dan disimpan dalam array bertipe struct Pemain. Proses ini memastikan bahwa setiap pemain memiliki identitas yang unik dan datanya tersimpan untuk diproses pada tahap selanjutnya.

Proses permainan dengan tebakan acak 1-100



```
Tebak angka 1-100 (ketik -1 untuk nyerah):
Tebakan: 66
Terlalu kecil!
Tebakan: 78
Terlalu besar!
Tebakan: 67
Terlalu kecil!
Tebakan: 73
Terlalu kecil!
Tebakan: 75
Benar!
```

Gambar 7. Input Angka Kartu

Setiap pemain akan bergiliran menebak angka rahasia yang telah diacak oleh program secara otomatis dalam rentang 1 sampai 100. Pemain diberikan petunjuk apakah tebakan mereka terlalu besar atau terlalu kecil. Proses ini berlangsung hingga pemain berhasil menebak angka dengan benar. Jumlah percobaan yang dibutuhkan akan dicatat dan disimpan untuk perhitungan skor.

Penilaian otomatis dan pengurutan skor

```
== Hasil Permainan ==
carol - Percobaan: 5, Skor: 80
saraa - Percobaan: 7, Skor: 70
```

Gambar 8. Hasil Permainan

Setelah semua pemain menyelesaikan giliran masing-masing, program secara otomatis menghitung skor berdasarkan jumlah percobaan yang dilakukan. Skor tertinggi diberikan kepada pemain dengan jumlah percobaan paling sedikit. Rumus yang digunakan untuk menghitung skor adalah  $100 - (\text{percobaan} - 1) \times 5$ , di mana nilai skor akan semakin rendah jika jumlah percobaan meningkat. Fungsi ini diimplementasikan secara rekursif untuk menunjukkan penerapan konsep rekursi dalam program.

Setelah skor dihitung, data pemain diurutkan menggunakan metode **Bubble Sort** berdasarkan jumlah percobaan secara ascending (dari yang terbaik ke terburuk). Pengurutan ini bertujuan untuk menampilkan peringkat pemain secara adil sesuai performa masing-masing. Bubble Sort dipilih karena mudah diimplementasikan dan cukup efisien untuk data berjumlah kecil.

Pencarian nama pemain

```
Cari nama pemain: saraa
Ditemukan! Skor 70, Percobaan 7
```

Gambar 9. Fitur Pencarian Nama Pemain

Fitur pencarian disediakan agar pengguna dapat melihat kembali skor dan data salah satu pemain tertentu. Program akan meminta input nama yang ingin dicari, lalu melakukan pencarian menggunakan algoritma **Linear Search**. Jika nama ditemukan, informasi lengkap seperti jumlah percobaan dan skor ditampilkan. Jika tidak ditemukan, program akan memberikan pesan bahwa data tidak tersedia.

## 2.6 Flowchart Alur Program



Gambar 10. Flowchart Alur Program

Gambar 10 di atas menunjukkan bahwa program dimulai dari input jumlah dan nama pemain, dilanjutkan dengan proses permainan, perhitungan skor menggunakan fungsi rekursif, pengurutan dengan Bubble Sort, pencarian nama pemain, hingga menampilkan peringkat. Diagram ini membantu memahami alur program secara menyeluruh.

## 3. PEMBAHASAN HASIL

Program Tebak Angka Multipemain berbasis C++ yang dikembangkan berhasil dijalankan dengan baik tanpa terdapat kesalahan fungsional. Seluruh fitur utama yang dirancang telah diimplementasikan dan diuji, mulai dari input data pemain, pelaksanaan permainan, perhitungan skor, pengurutan, hingga pencarian data.

Berdasarkan hasil pengujian yang telah dilakukan, diperoleh beberapa temuan penting sebagai berikut:

- Setiap pemain dapat bermain secara bergiliran. Sistem permainan menggunakan mekanisme looping pada array Pemain, di mana setiap indeks mewakili satu pemain yang mendapat giliran secara berurutan. Hal ini menunjukkan bahwa kontrol alur permainan berjalan sesuai rencana, dan tidak ada giliran yang terlewat atau terduplicasi.
- Pointer memungkinkan modifikasi data pemain dari fungsi mainGame(). Penggunaan pointer sebagai parameter pada fungsi mainGame() memberikan kemampuan untuk mengubah data secara langsung tanpa mengembalikan nilai. Hal ini menunjukkan pemahaman penggunaan parameter by reference dalam C++ untuk menghemat memori dan meningkatkan efisiensi akses data.
- Sorting dengan Bubble Sort bekerja efisien untuk data berukuran kecil. Meskipun Bubble Sort memiliki kompleksitas waktu  $O(n^2)$ , hasil pengujian menunjukkan bahwa untuk jumlah data pemain yang relatif kecil (sekitar 3–10 orang), waktu proses pengurutan berjalan cepat dan efisien. Urutan pemain berdasarkan jumlah percobaan berhasil ditampilkan secara ascending, yang juga mencerminkan urutan performa.
- Fungsi hitungSkor() menghasilkan skor sesuai logika yang telah ditentukan. Rumus skor =  $100 - (\text{percobaan} - 1) \times 5$  telah diterapkan menggunakan pendekatan rekursif. Fungsi



ini mengembalikan nilai skor yang berbanding terbalik dengan jumlah percobaan, memberikan insentif bagi pemain yang dapat menebak angka dengan lebih cepat. Pengujian dengan berbagai skenario menunjukkan bahwa skor dihitung secara akurat dan tidak pernah bernilai negatif, karena terdapat pengamanan untuk nilai minimum skor.

- Fitur pencarian nama pemain berjalan dengan akurat.

Fungsi pencarian menggunakan algoritma Linear Search, yang memungkinkan pengguna untuk mencari data berdasarkan nama. Ketika nama ditemukan, informasi seperti jumlah percobaan dan skor akan ditampilkan. Jika nama tidak ditemukan, program memberikan pesan yang informatif. Fitur ini menambah aspek interaktif pada program dan meningkatkan pengalaman pengguna.

Selain itu, program menampilkan antarmuka berbasis teks (terminal) yang sederhana namun informatif. Output yang ditampilkan mudah dibaca dan interaksi pengguna cukup intuitif. Setelah permainan berakhir, seluruh data pemain ditampilkan dalam bentuk tabel, memungkinkan pengguna untuk melihat perbandingan skor dan jumlah percobaan antar pemain.

Secara keseluruhan, implementasi game edukatif ini tidak hanya berhasil memenuhi aspek teknis, tetapi juga mendukung proses pembelajaran konsep dasar pemrograman dalam konteks yang menyenangkan dan aplikatif.

#### 4. KESIMPULAN

Permainan edukatif Tebak Angka berbasis C++ ini berhasil mengimplementasikan berbagai konsep dasar pemrograman seperti pointer, array, struct, fungsi prosedural dan rekursif, serta algoritma sorting dan searching. Program dirancang dengan pendekatan modular dan interaktif sehingga cocok digunakan sebagai media pembelajaran logika pemrograman di lingkungan akademik. Selain memberikan pengalaman bermain yang menyenangkan, program ini juga membantu pengguna memahami penerapan konsep pemrograman secara langsung. Ke depannya, pengembangan program dapat mencakup penyimpanan data ke file, leaderboard permanen, dan tampilan antarmuka grafis agar lebih menarik dan fungsional.

#### UCAPAN TERIMA KASIH

Penulis menyampaikan penghargaan yang sebesar-besarnya kepada:

**Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro PSDKU Kediri**, yang telah memberikan dukungan sarana dan prasarana, serta kesempatan untuk menyalurkan ide dan kreativitas dalam bentuk karya ilmiah terapan berbasis teknologi.

#### REFERENCES

- [1] I. Ice *et al.*, ‘‘PENGENALAN PEMROGRAMAN DASAR DUNIA KODING DENGAN C++.’’
- [2] Adawiyah Ritonga and Yahfizham Yahfizham, ‘‘Studi Literatur Perbandingan Bahasa Pemrograman C++ dan Bahasa Pemrograman Python pada Algoritma Pemrograman,’’ *Jurnal Teknik Informatika dan Teknologi Informasi*, vol. 3, no. 3, pp. 56–63, Nov. 2023, doi: 10.55606/jutiti.v3i3.2863.
- [3] B. Tahun, ‘‘MIFORTEKH (Jurnal Manajemen Informatika & Teknologi) Integrasi Algoritma Rekursif pada Pemrosesan Data Multilevel untuk Aplikasi Berbasis AI,’’ vol. XX, [Online]. Available: <https://https://journal.stiestekom.ac.id/index.php/mifortekh>
- [4] R. Resti, R. A. Wati, S. Ma’Arif, and S. Syarifuddin, ‘‘Pemanfaatan Media Pembelajaran Berbasis Teknologi sebagai Alat Untuk Meningkatkan Kemampuan Literasi Digital Siswa Sekolah Dasar,’’ *Al Madrasah Jurnal Pendidikan Madrasah Ibtidaiyah*, vol. 8, no. 3, p. 1145, Jul. 2024, doi: 10.35931/am.v8i3.3563.
- [5] H. Saputra, D. Darwis, E. Febrianto, P. Matematika, F. Sastra, and I. Pendidikan, ‘‘RANCANG BANGUN APLIKASI GAME MATEMATIKA UNTUK PENYANDANG TUNAGRAHITA BERBASIS MOBILE,’’ *Jurnal Komputer dan Informatika*, vol. 15, pp. 73–83, 2020.
- [6] A. Musrifah, I. Nursida, and F. Setiawati Sulaeman, ‘‘GAME EDUKASI BAGI ANAK ATTENTION DEFICIT HYPERACTIVE DISORDER (ADHD) BERBASIS ANDROID’’, doi: 10.31949/infotech.v8I2.3464.



- [7] H. Mayatopani and N. Handayani, "DESIGN IMPLEMENTASI KECERDASAAN BUATAN PADA CHATBOT LINE DALAM PERMAINAN TEBAK ANGKA," 2019. [Online]. Available: <https://developers.line.me/>
- [8] M. Literasi *et al.*, "Pelatihan Pemrograman Dasar Python," 2025.
- [9] G. Ngurah, A. Krisnawan, S. S. Stmik, S. Bali, and J. Raya Puputan, "Konferensi Nasional Sistem & Informatika," 2015.
- [10] I. Ramadhana and B. Sujatmiko, "PENGEMBANGAN APLIKASI KAMUS BAHASA PEMROGRAMAN C++ BERBASIS ANDROID UNTUK MENINGKATKAN KOMPETENSI KOGNITIF MATA KULIAH STRUKTUR DATA."
- [11] M. Sukarnih Putri and E. Budiarti, "PENGEMBANGAN GAME EDUKATIF TEBAK GAMBAR DAN HURUF UNTUK PERKEMBANGAN KOGNITIF ANAK USIA DINI."
- [12] M. Yusril Pratama and Khusnul Khotimah, "Perancangan dan Implementasi Game Edukasi Tebak Gambar dan Angka Sebagai Media Pembelajaran Calistung (Baca, Tulis, dan Berhitung) Taman Kanak-Kanak Berbasis Android," *Sienna*, vol. 5, no. 1, pp. 52–65, Jul. 2024, doi: 10.47637/sienna.v5i1.1158.
- [13] R. W. Arifin and D. Setiyadi, "Algoritma Metode Pengurutan Bubble Sort dan Quick Sort Dalam Bahasa Pemrograman C++," *INFORMATION SYSTEM FOR EDUCATORS AND PROFESSIONALS*, vol. 4, no. 2, pp. 178–187, 2020.
- [14] A. Hendra Saptadi and D. Windi Sari, "ANALISIS ALGORITMA INSERTION SORT, MERGE SORT DAN IMPLEMENTASINYA DALAM BAHASA PEMROGRAMAN C++," 2012.
- [15] D. Oleh, "PERANCANGAN PROGRAM PENGHITUNG JUMLAH KENDARAAN DI LINTASAN JALAN RAYA SATU ARAH MENGGUNAKAN BAHASA PEMROGRAMAN C++ DENGAN PUSTAKA OPENCV Publikasi Jurnal Skripsi."