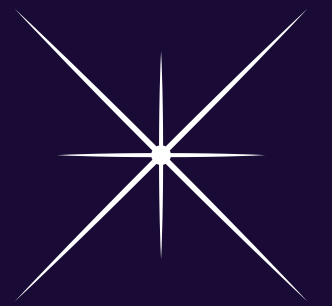
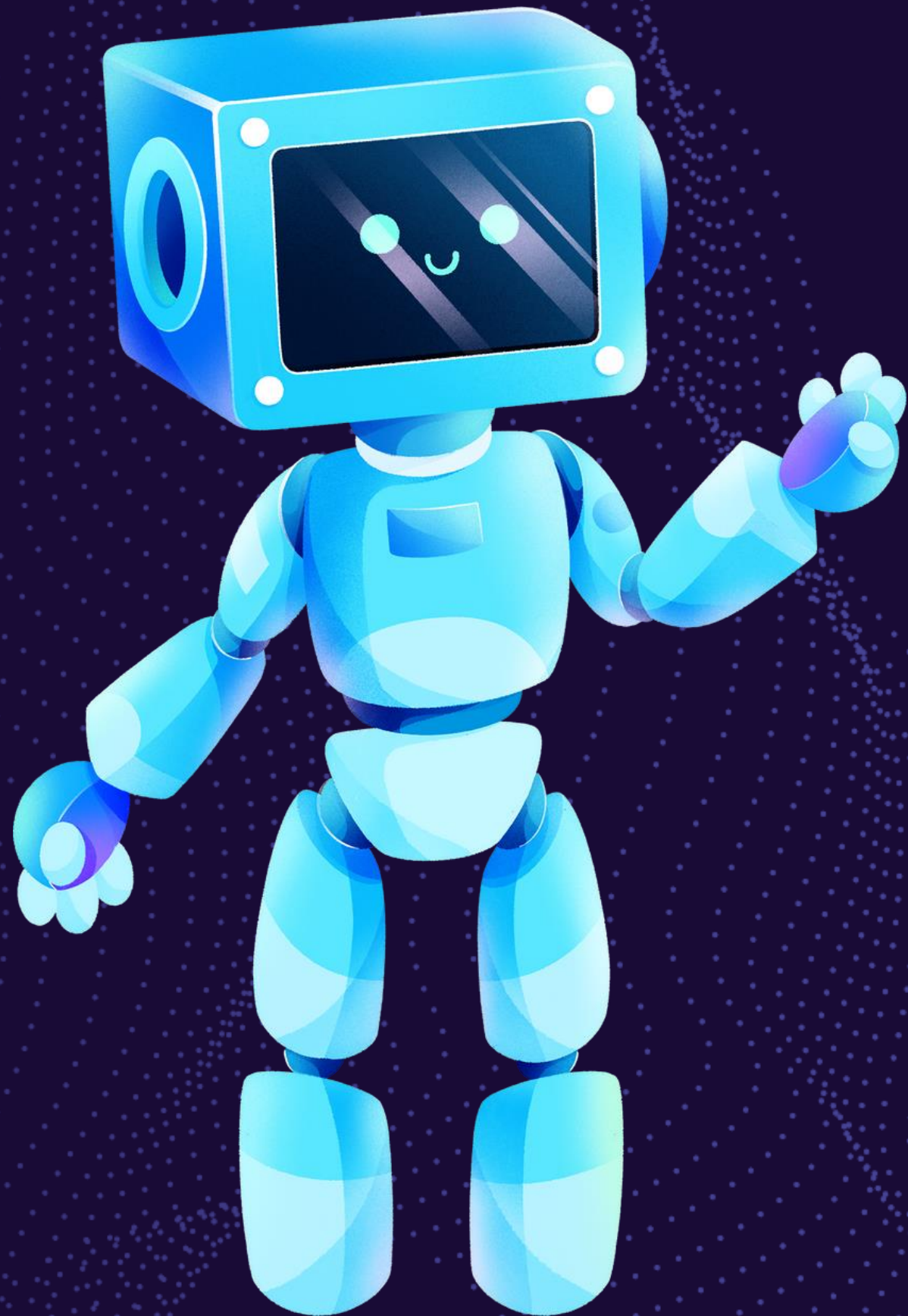


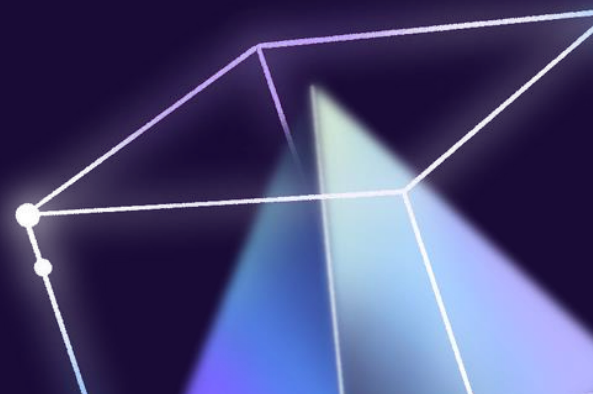
STRUKTUR PERULANGAN DAN ARRAY DALAM BAHASA PEMROGRAMAN C++

Kelompok 3



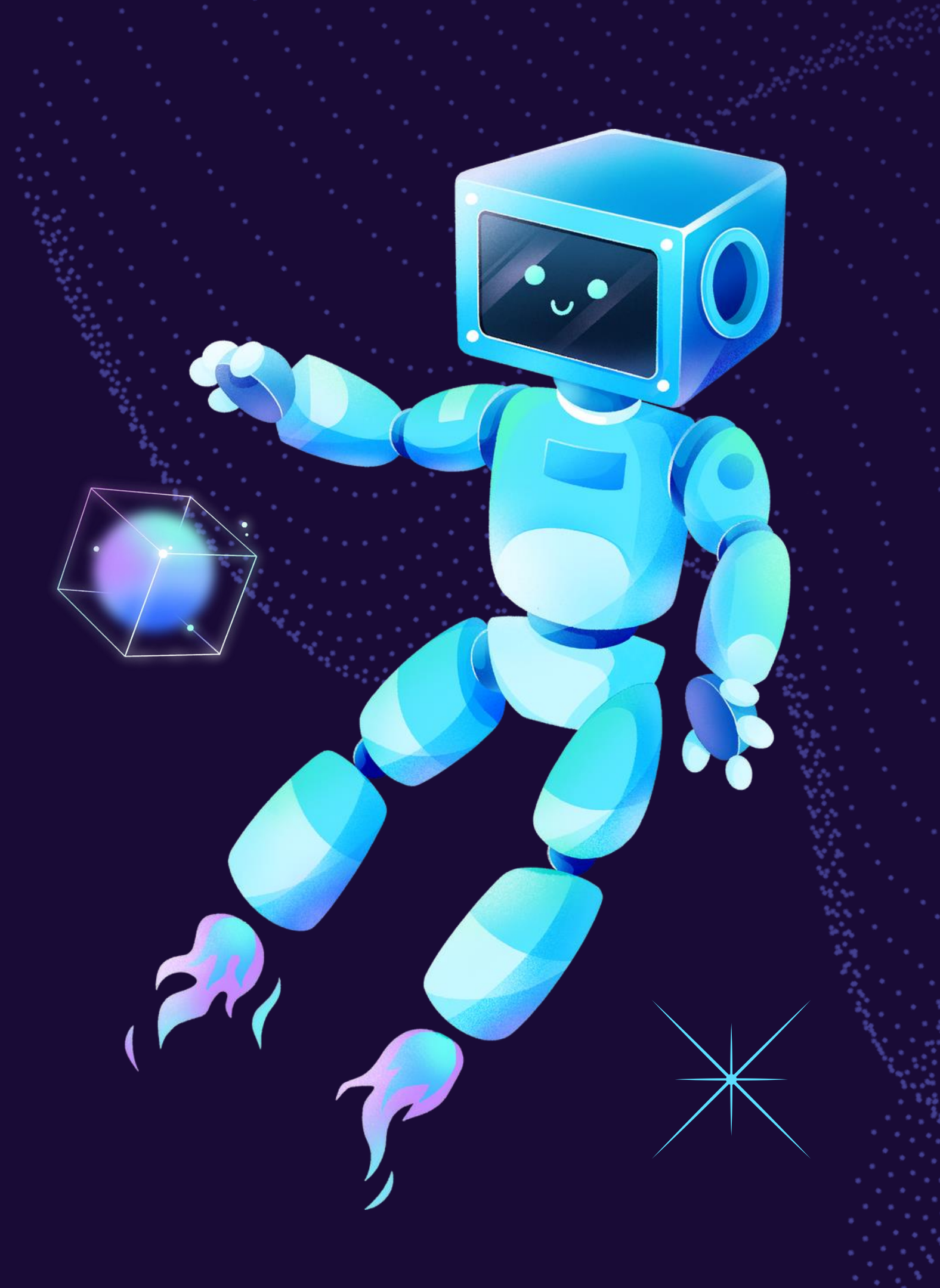
Kelompok 3

- Adi Abdul Riadi
- Asep Haryana Saputra
- Muhammad Rizal Nurfirdaus



Pengertian Struktur Perulangan

Struktur perulangan adalah salah satu konsep dasar dalam pemrograman komputer yang memungkinkan sebuah program menjalankan serangkaian instruksi secara berulang-ulang selama kondisi tertentu terpenuhi. Tujuan dari struktur perulangan adalah untuk mengotomatisasi tugas-tugas yang perlu diulang tanpa harus menulis ulang kode secara manual.

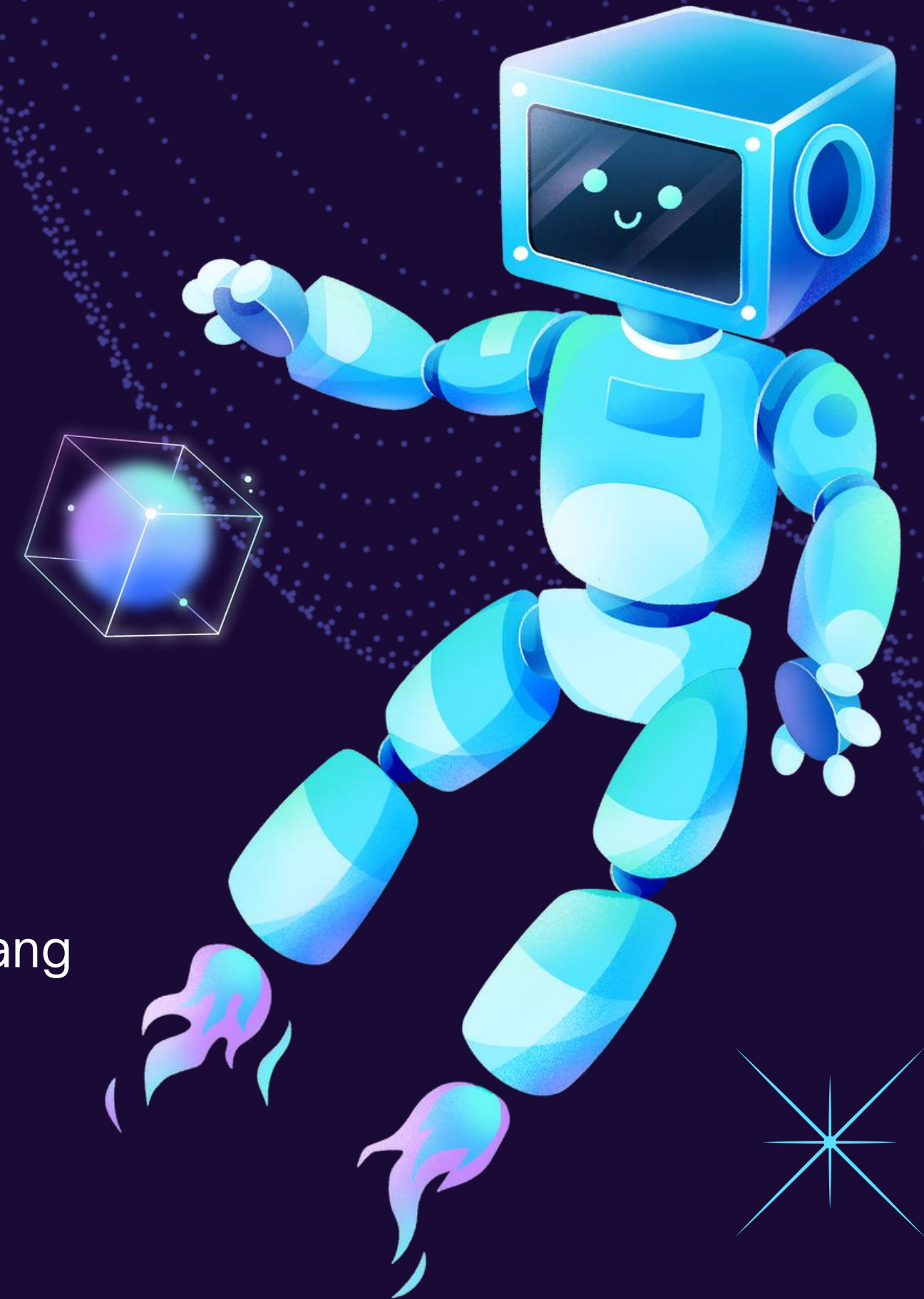


a. Bagaimana cara menggunakan struktur perulangan for dalam bahasa pemrograman C++?

Struktur perulangan dalam bahasa pemrograman C++ bertujuan agar programmer tidak perlu menuliskan perintah yang sama, cukup beberapa perintah saja. Ada tiga jenis perulangan dalam C++, yaitu For, While, dan Do While.

Perulangan For

Perulangan For biasa dipakai untuk mengulang sebuah proses yang sudah diketahui jumlahnya.



Berikut adalah langkah-langkah untuk menggunakan perulangan for dalam C++:

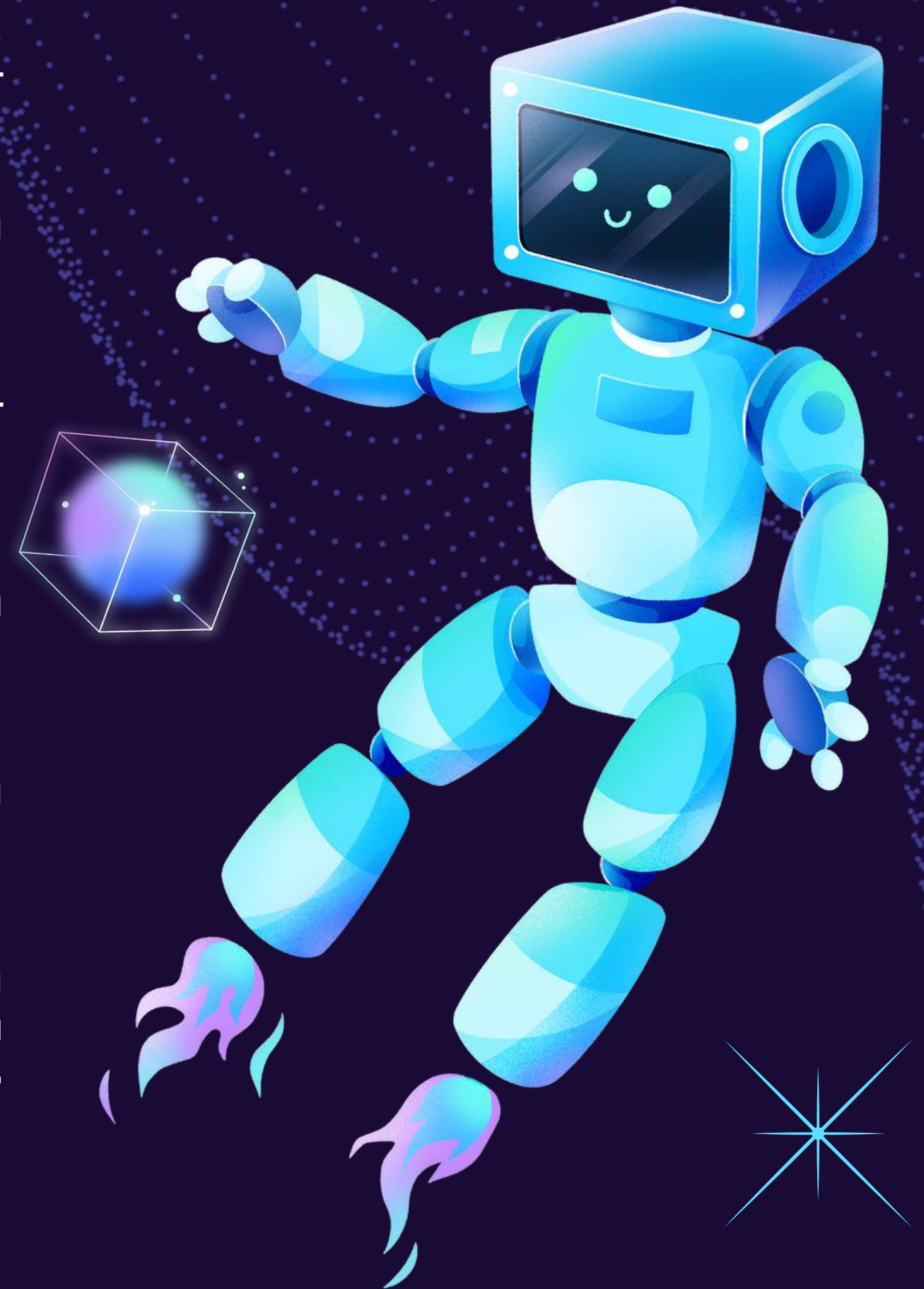
Inisialisasi: Pertama, kita perlu menginisialisasi variabel yang akan digunakan dalam perulangan. Biasanya, variabel ini digunakan sebagai penghitung jumlah perulangan. Contoh: `int i = 0;`

Kondisi: Selanjutnya, kita menentukan kondisi yang harus dipenuhi agar perulangan terus berjalan. Jika kondisi ini tidak lagi terpenuhi, maka perulangan akan berhenti. Contoh: `i < 10;`

Eksekusi Pernyataan: Jika kondisi terpenuhi, maka pernyataan dalam blok perulangan akan dieksekusi. Pernyataan ini bisa berupa satu atau lebih baris kode.

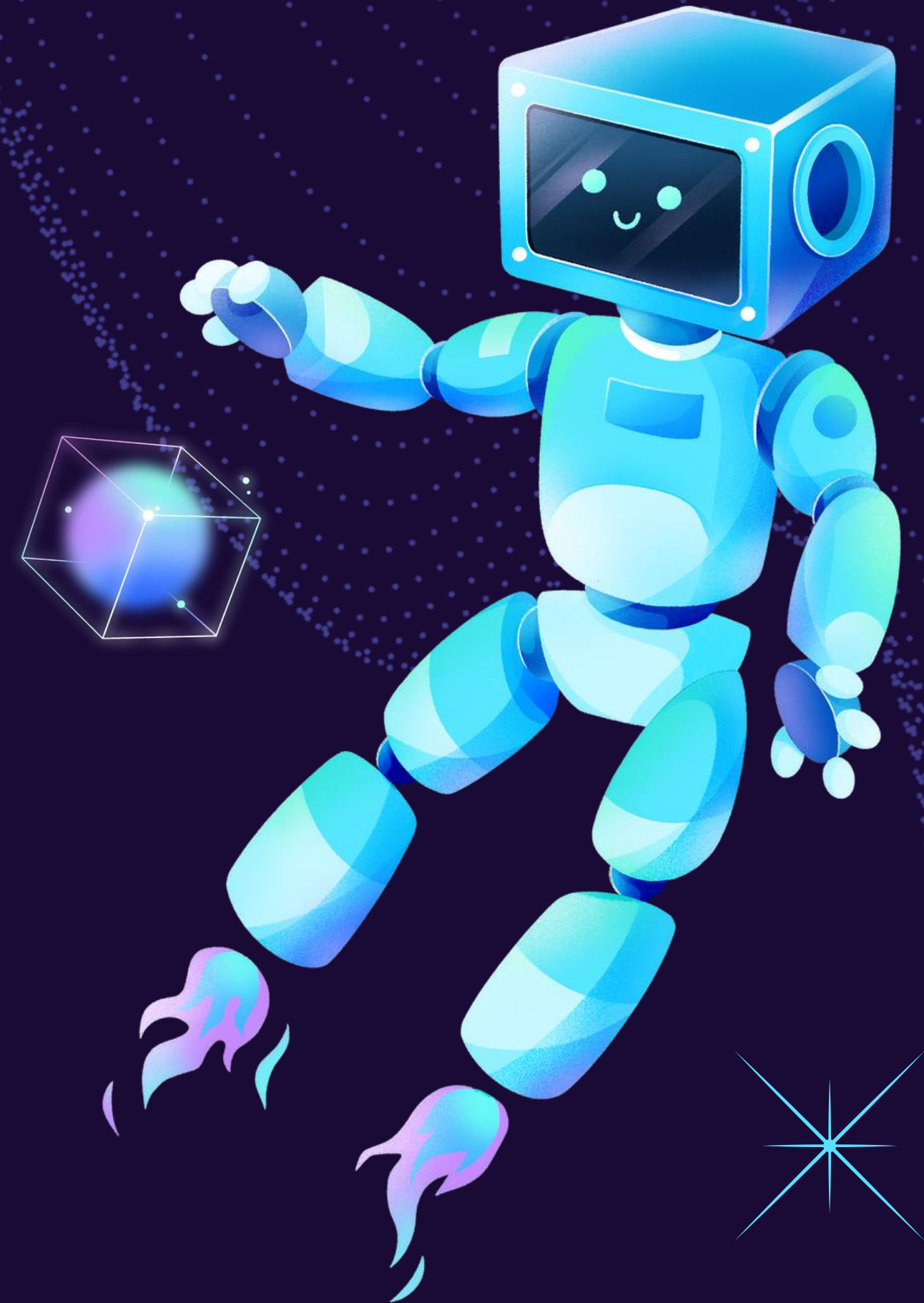
Modifikasi: Setelah pernyataan dieksekusi, variabel penghitung akan dimodifikasi. Modifikasi ini biasanya berupa penambahan atau pengurangan nilai variabel. Contoh: `i++`.

Pengecekan Kondisi: Setelah modifikasi, kondisi akan dicek kembali. Jika kondisi masih terpenuhi, maka pernyataan akan dieksekusi lagi dan variabel penghitung akan dimodifikasi lagi. Proses ini akan terus berulang sampai kondisi tidak lagi terpenuhi.



Notasi Algoritma

```
for (inisialisasi; kondisi; modifier) {  
    // Pernyataan yang dieksekusi;  
}
```

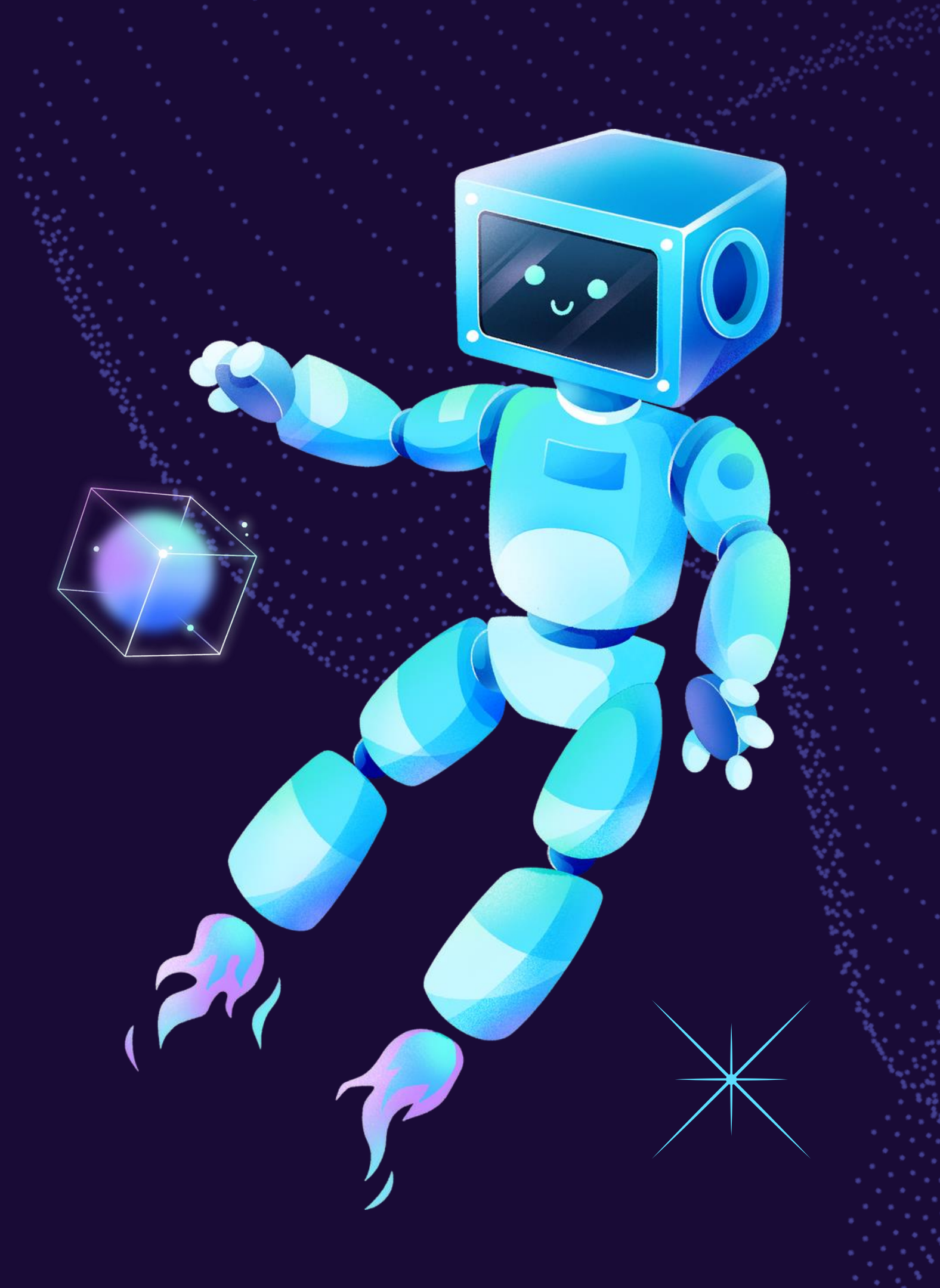


Berikut adalah contoh penulisan perulangan For:

```
#include <iostream>
using namespace std;

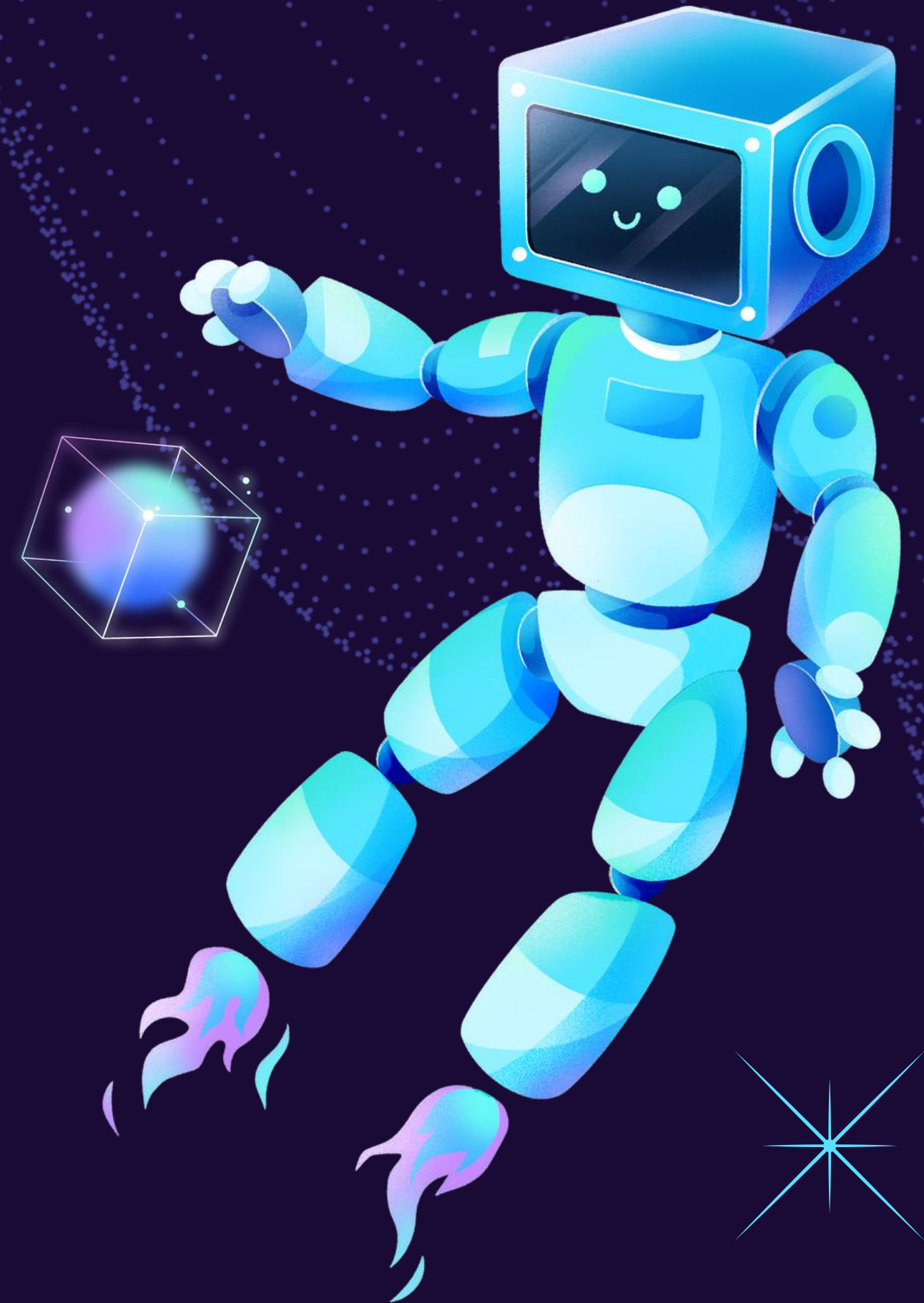
int main() {
    for (int i = 0; i < 10; i++) {
        cout << i << endl;
    }

    return 0;
}
```



Hasil Run Program:

1
2
3
4
5
6
7
8
9
10



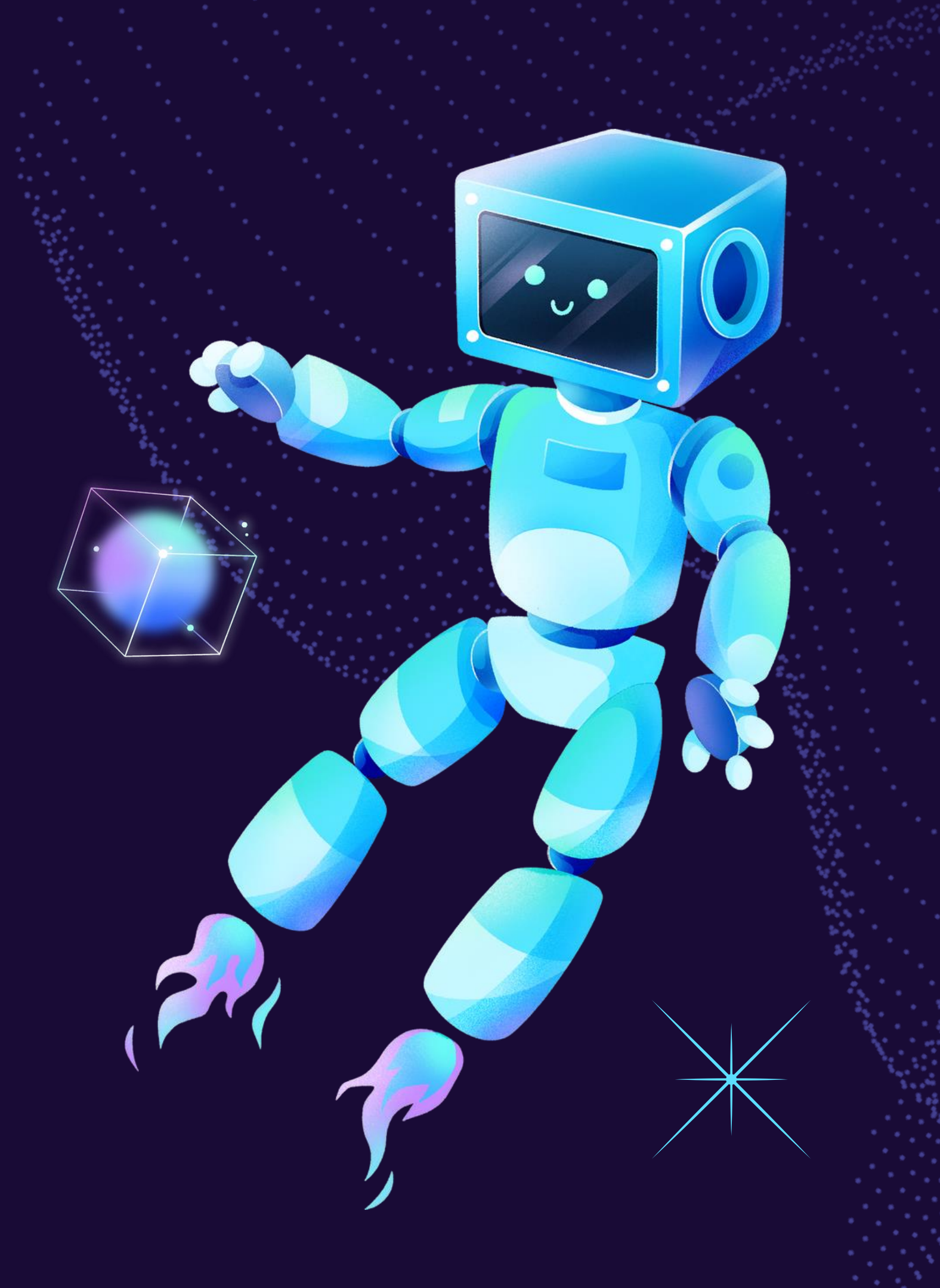
b. Bagaimana cara menggunakan struktur perulangan while dalam bahasa

pemrograman C++?

Perulangan while

Perulangan While banyak dipakai untuk program yang lebih terstruktur, namun

berapa jumlah perulangannya belum diketahui.



Berikut adalah langkah-langkah untuk menggunakan perulangan while dalam C++:

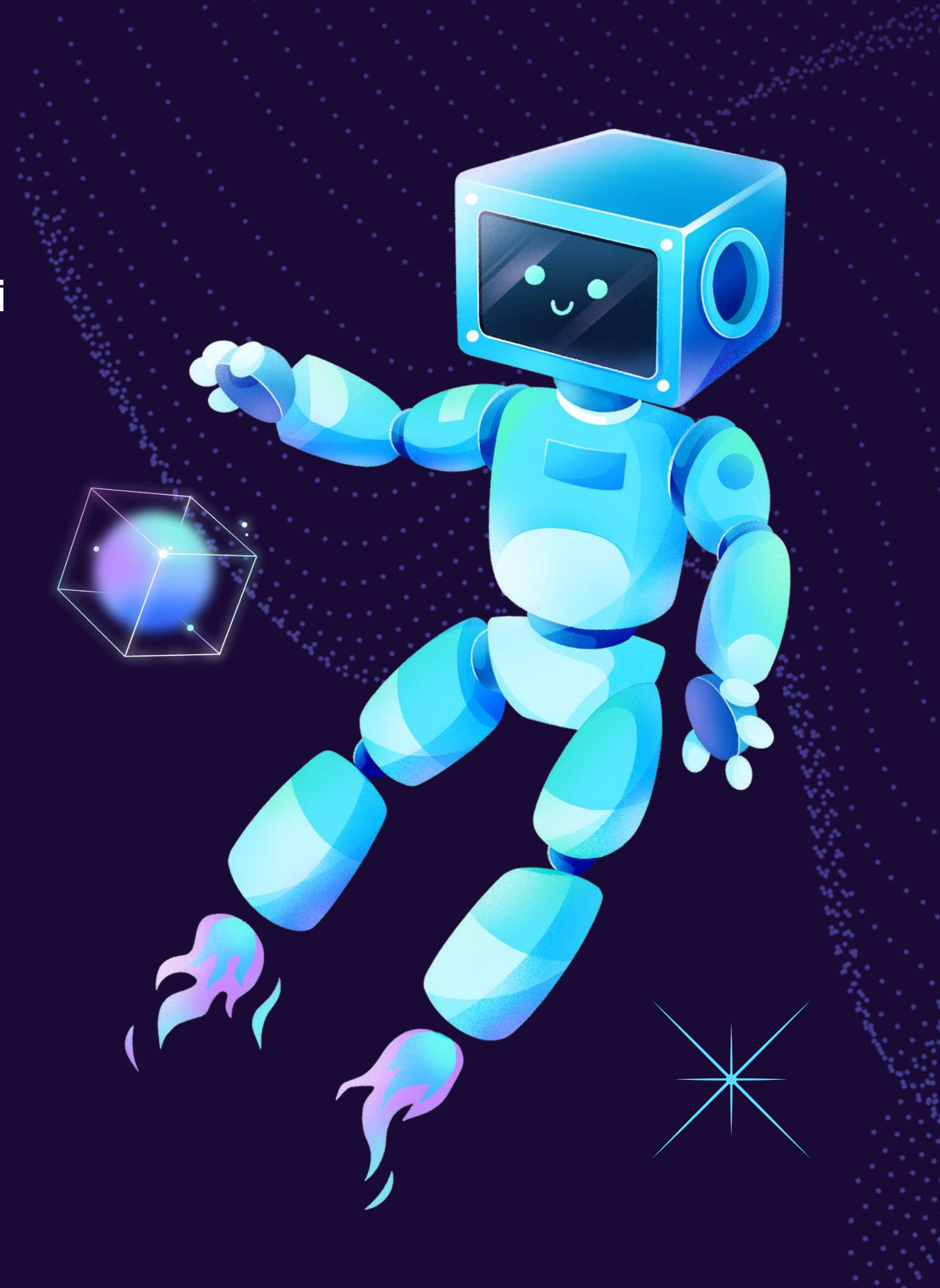
Inisialisasi: Pertama, kita perlu menginisialisasi variabel yang akan digunakan dalam perulangan. Variabel ini biasanya digunakan sebagai penghitung jumlah perulangan. Contoh: `int i = 1;`

Kondisi: Selanjutnya, kita menentukan kondisi yang harus dipenuhi agar perulangan terus berjalan. Jika kondisi ini tidak lagi terpenuhi, maka perulangan akan berhenti. Contoh: `i <= 5;`

Eksekusi Pernyataan: Jika kondisi terpenuhi, maka pernyataan dalam blok perulangan akan dieksekusi. Pernyataan ini bisa berupa satu atau lebih baris kode.

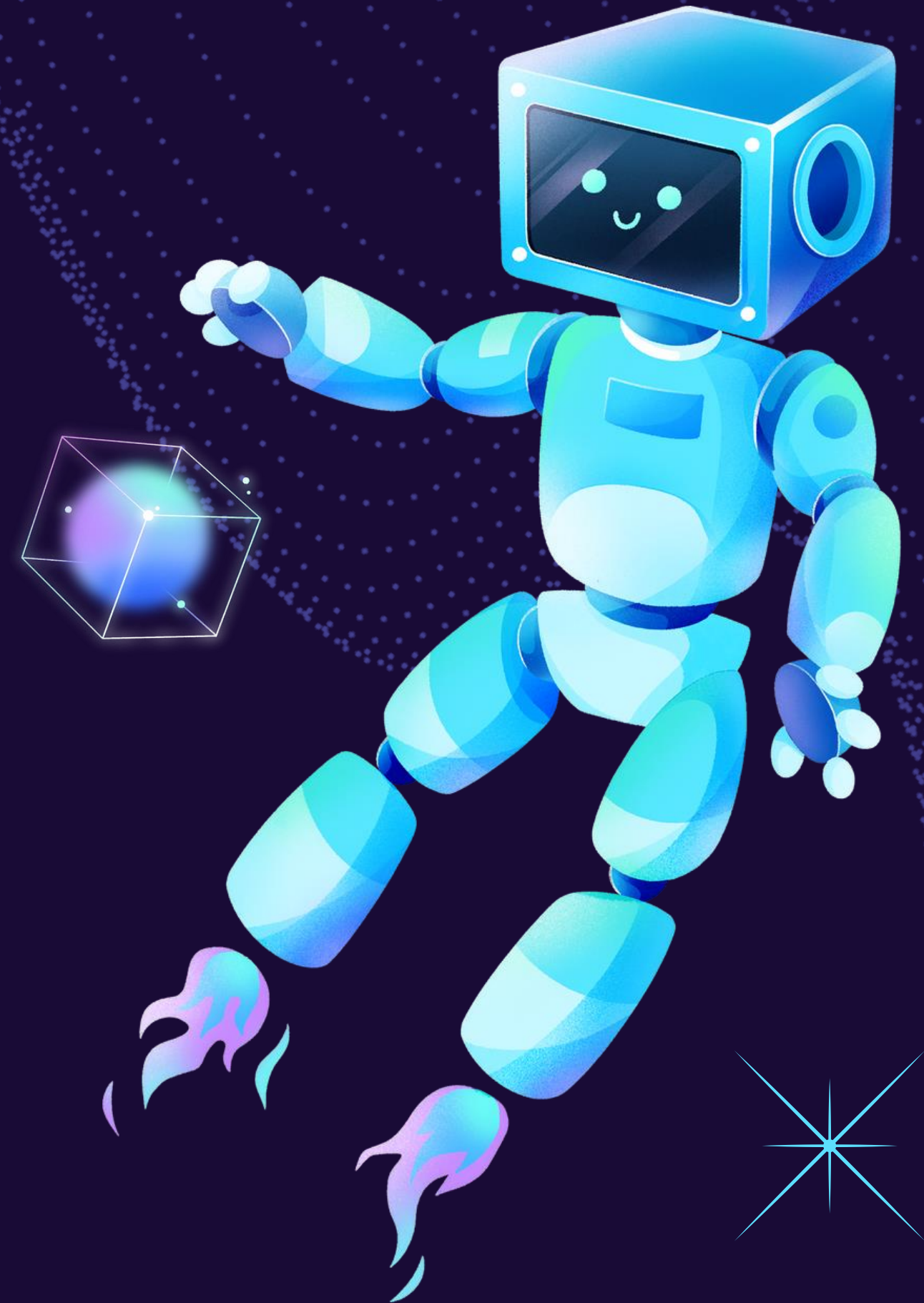
Modifikasi: Setelah pernyataan dieksekusi, variabel penghitung akan dimodifikasi. Modifikasi ini biasanya berupa penambahan atau pengurangan nilai variabel. Contoh: `i++;`

Pengecekan Kondisi: Setelah modifikasi, kondisi akan dicek kembali. Jika kondisi masih terpenuhi, maka pernyataan akan dieksekusi lagi dan variabel penghitung akan dimodifikasi lagi. Proses ini akan terus berulang sampai kondisi tidak lagi terpenuhi.



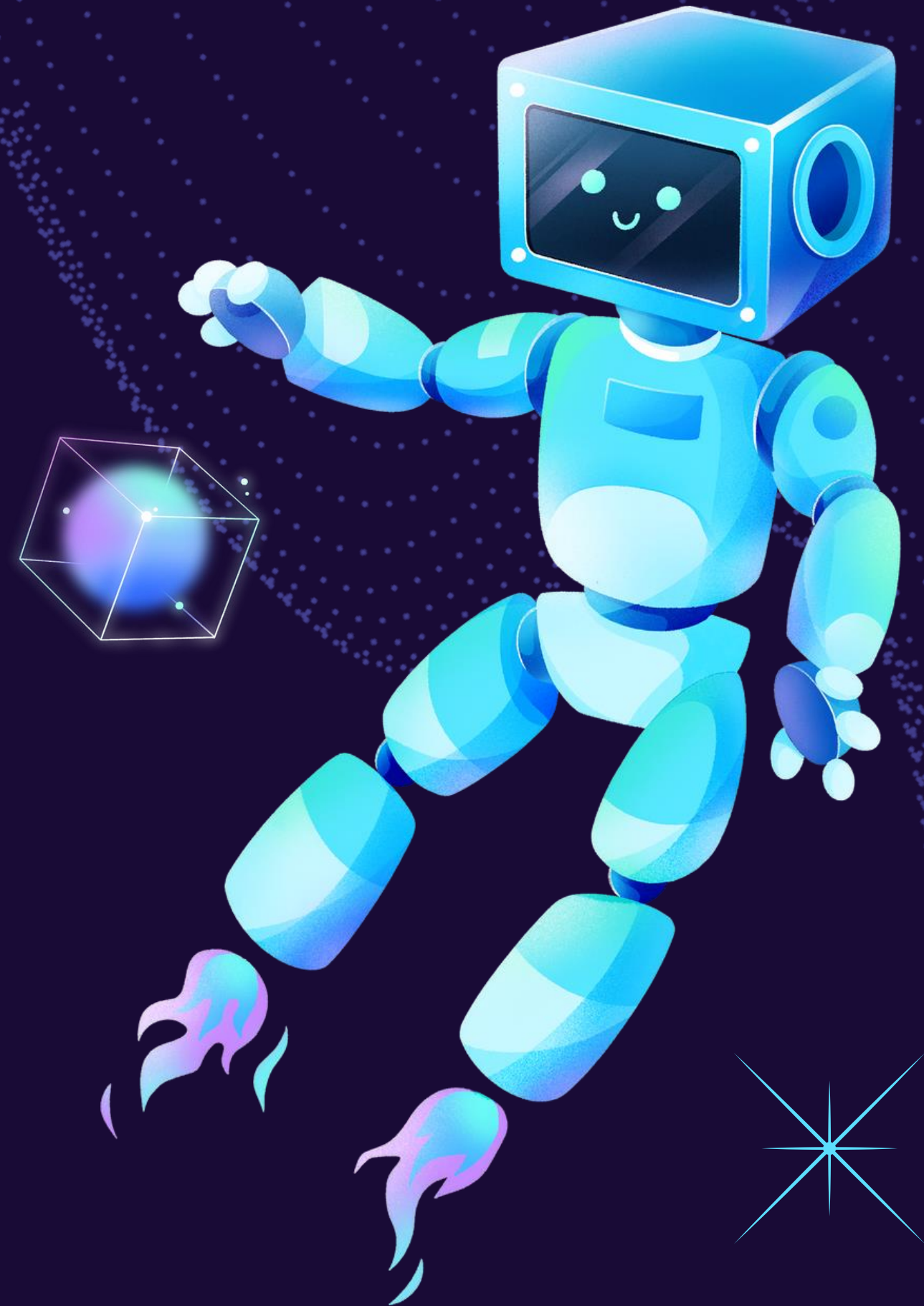
Notasi Algoritma

```
while (kondisi) {  
  // Eksekusi pernyataan;  
}
```



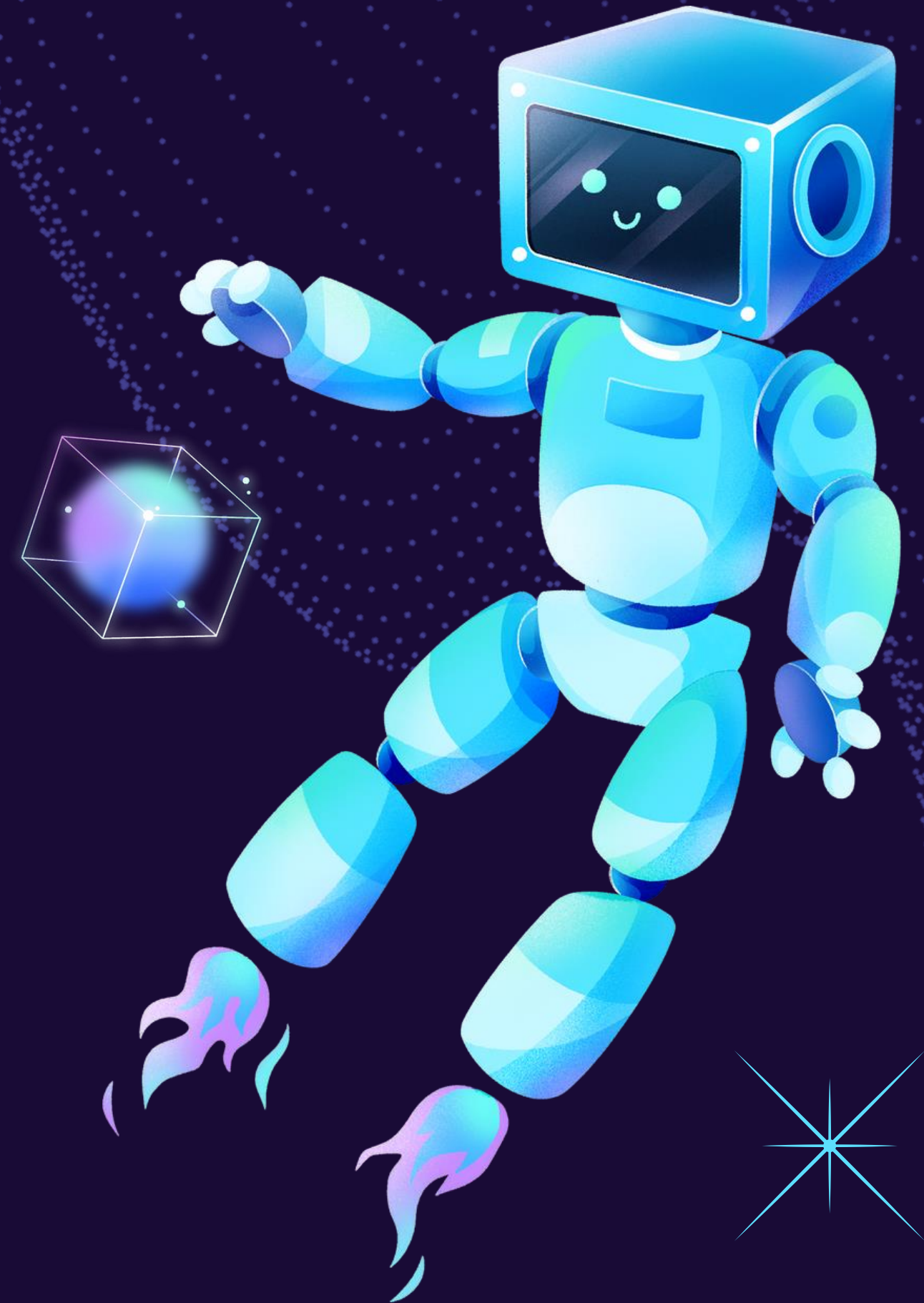
Berikut adalah contoh penulisan
perulangan While:

```
#include <iostream>
using namespace std;
int main() {
    int i = 0;
    while (i < 10) {
        cout << i << endl;
        i++;
    }
    return 0;
}
```



Hasil Run Program:

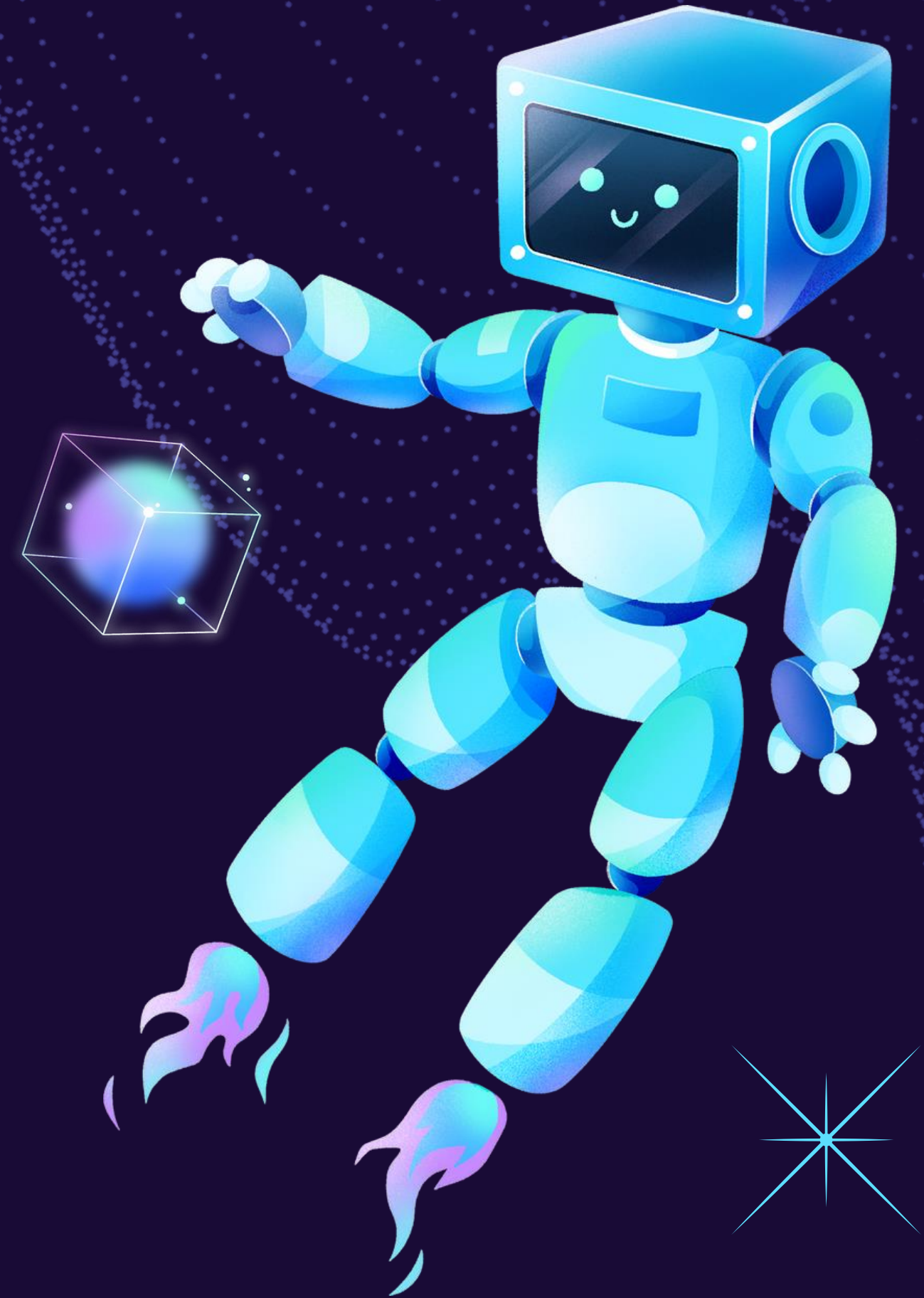
1
2
3
4
8
5
6
7
8
9



c. Bagaimana cara menggunakan struktur perulangan do while dalam bahasa pemrograman C++?

Perulangan do while

Perulangan Do While hampir sama dengan perulangan While, bedanya ada pada statement perulangan yang dilakukan terlebih dahulu baru kondisinya di cek.



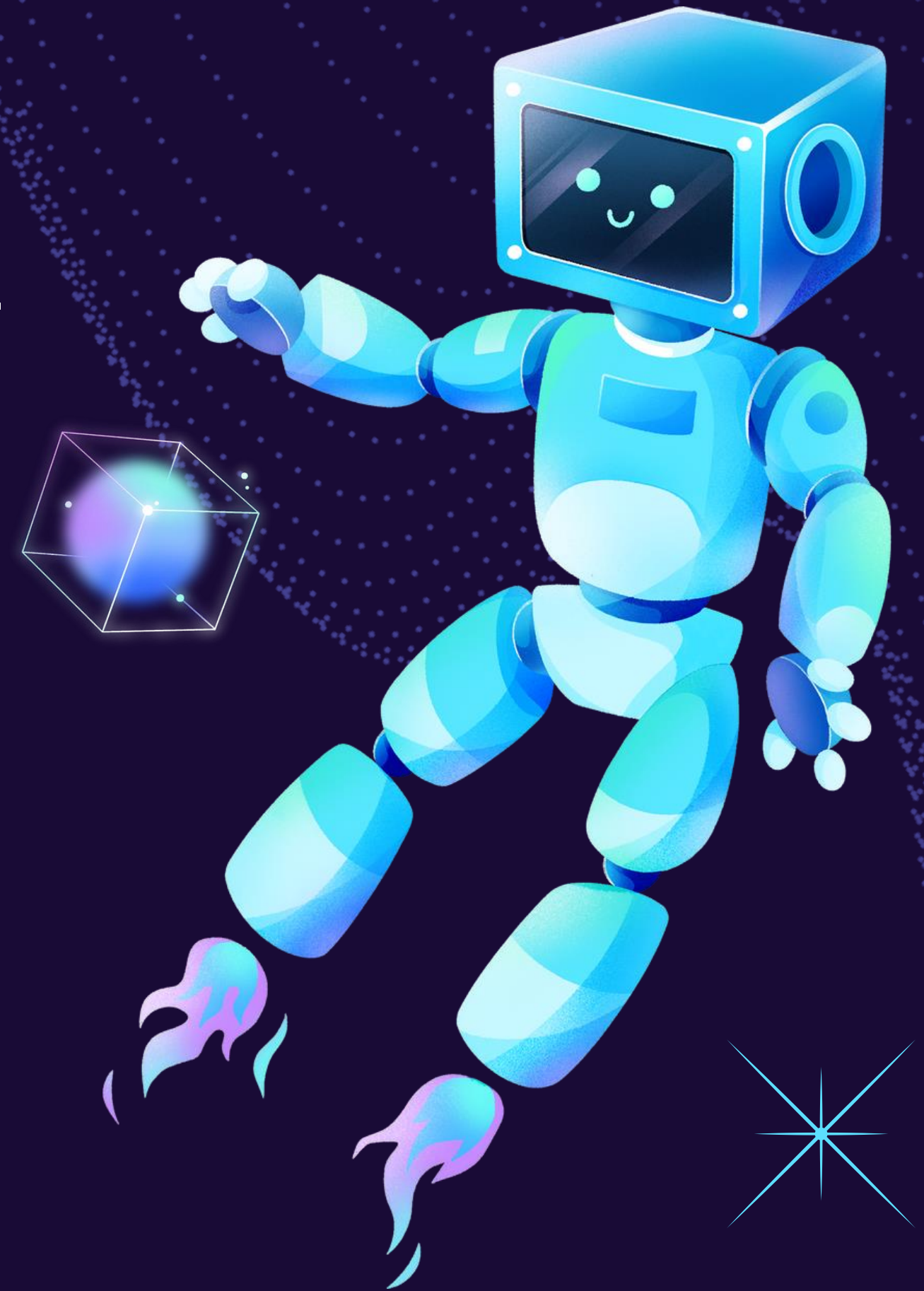
Berikut adalah langkah-langkah untuk menggunakan perulangan do while dalam C++:

Inisialisasi: Pertama, kita perlu menginisialisasi variabel yang akan digunakan dalam perulangan. Variabel ini biasanya digunakan sebagai penghitung jumlah perulangan. Contoh: `int a = 1;`

Eksekusi Pernyataan: Kemudian, pernyataan dalam blok perulangan akan dieksekusi. Pernyataan ini bisa berupa satu atau lebih baris kode.

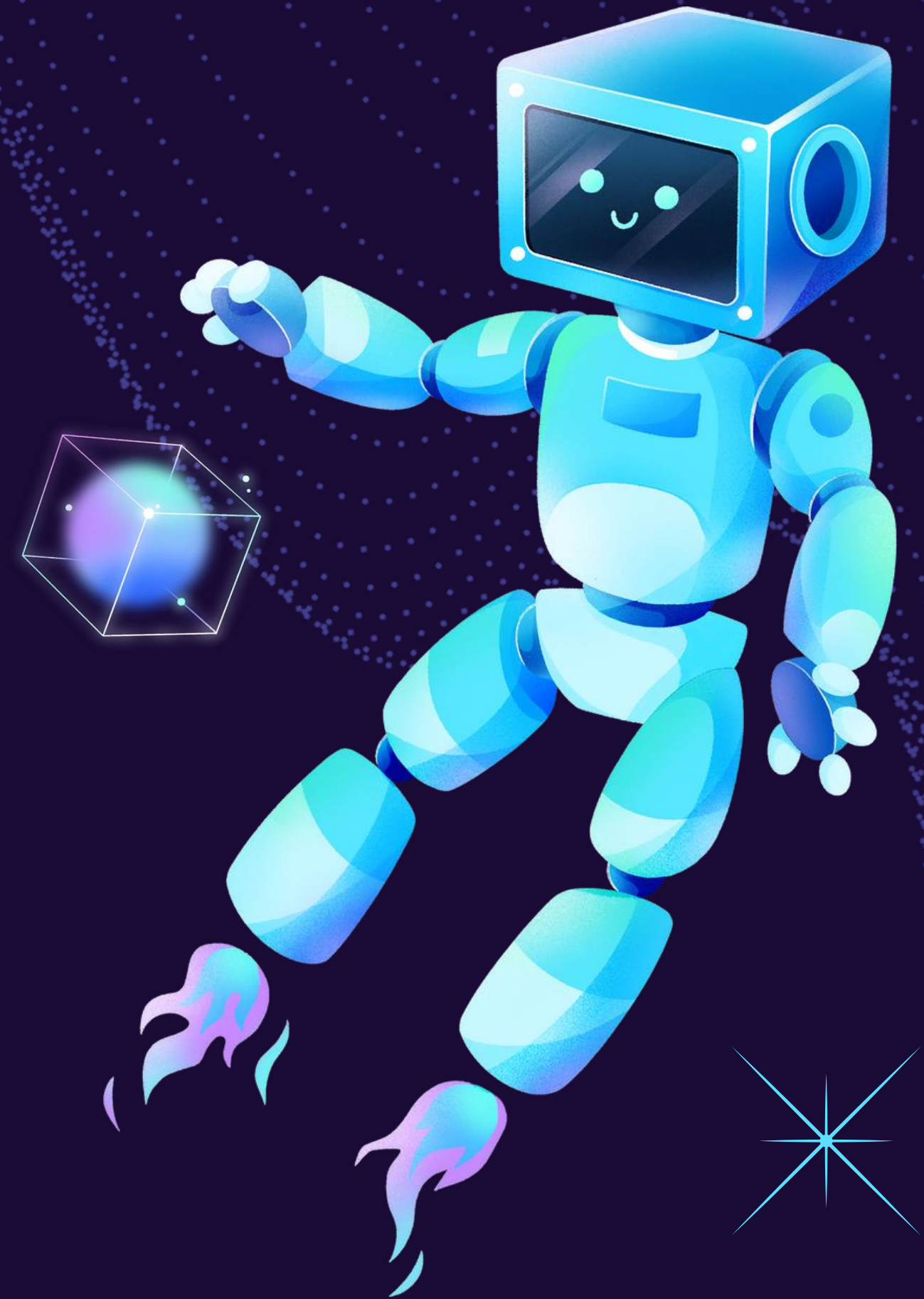
Modifikasi: Setelah pernyataan dieksekusi, variabel penghitung akan dimodifikasi. Modifikasi ini biasanya berupa penambahan atau pengurangan nilai variabel. Contoh: `a++;`

Pengecekan Kondisi: Setelah modifikasi, kondisi akan dicek. Jika kondisi masih terpenuhi, maka pernyataan akan dieksekusi lagi dan variabel penghitung akan dimodifikasi lagi. Proses ini akan terus berulang sampai kondisi tidak lagi terpenuhi



Notasi Algoritma

```
do {  
  // Pernyataan yang dieksekusi;  
} while (kondisi);
```

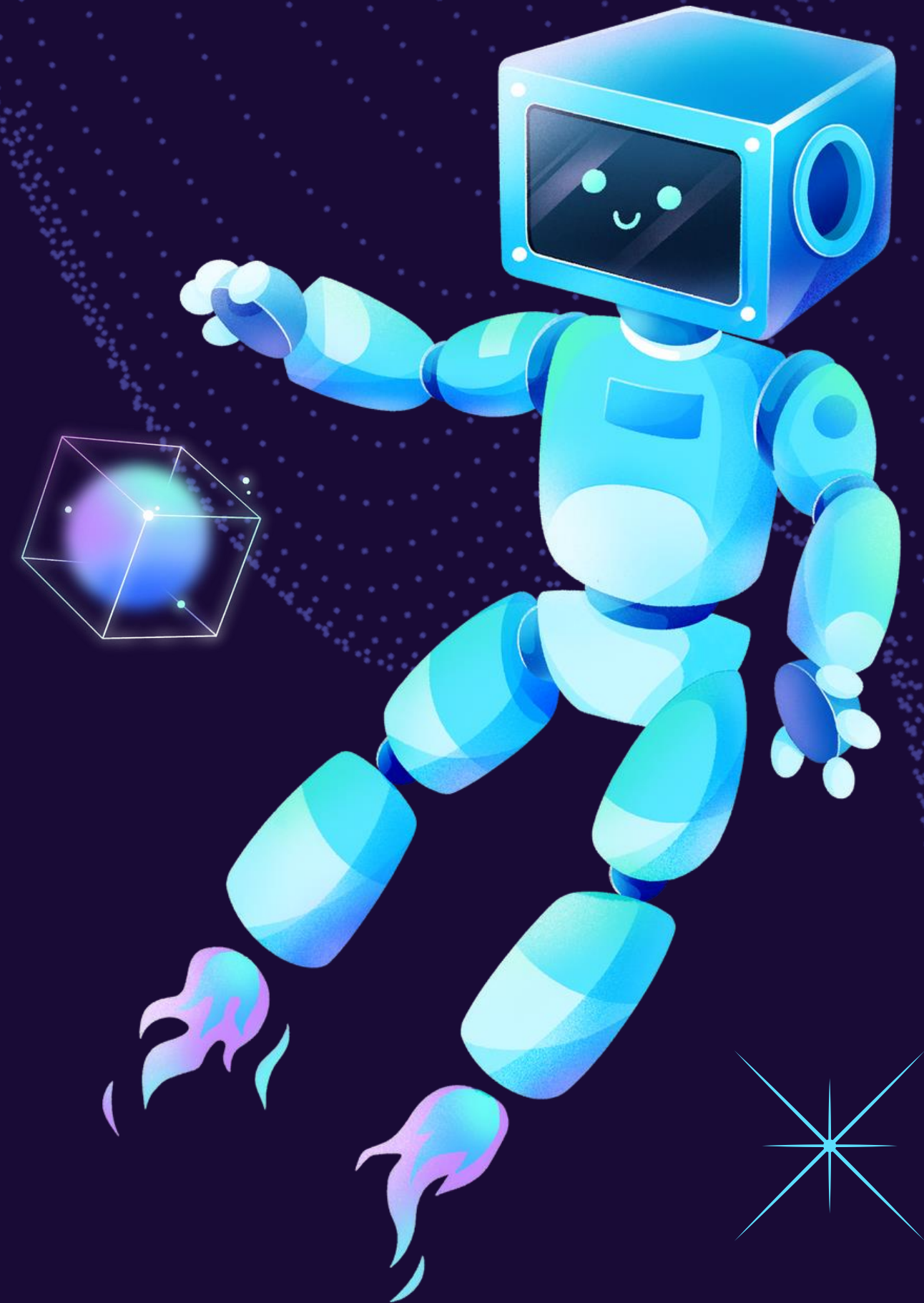


Berikut adalah contoh penulisan perulangan Do While:

```
#include <iostream>
using namespace std;
```

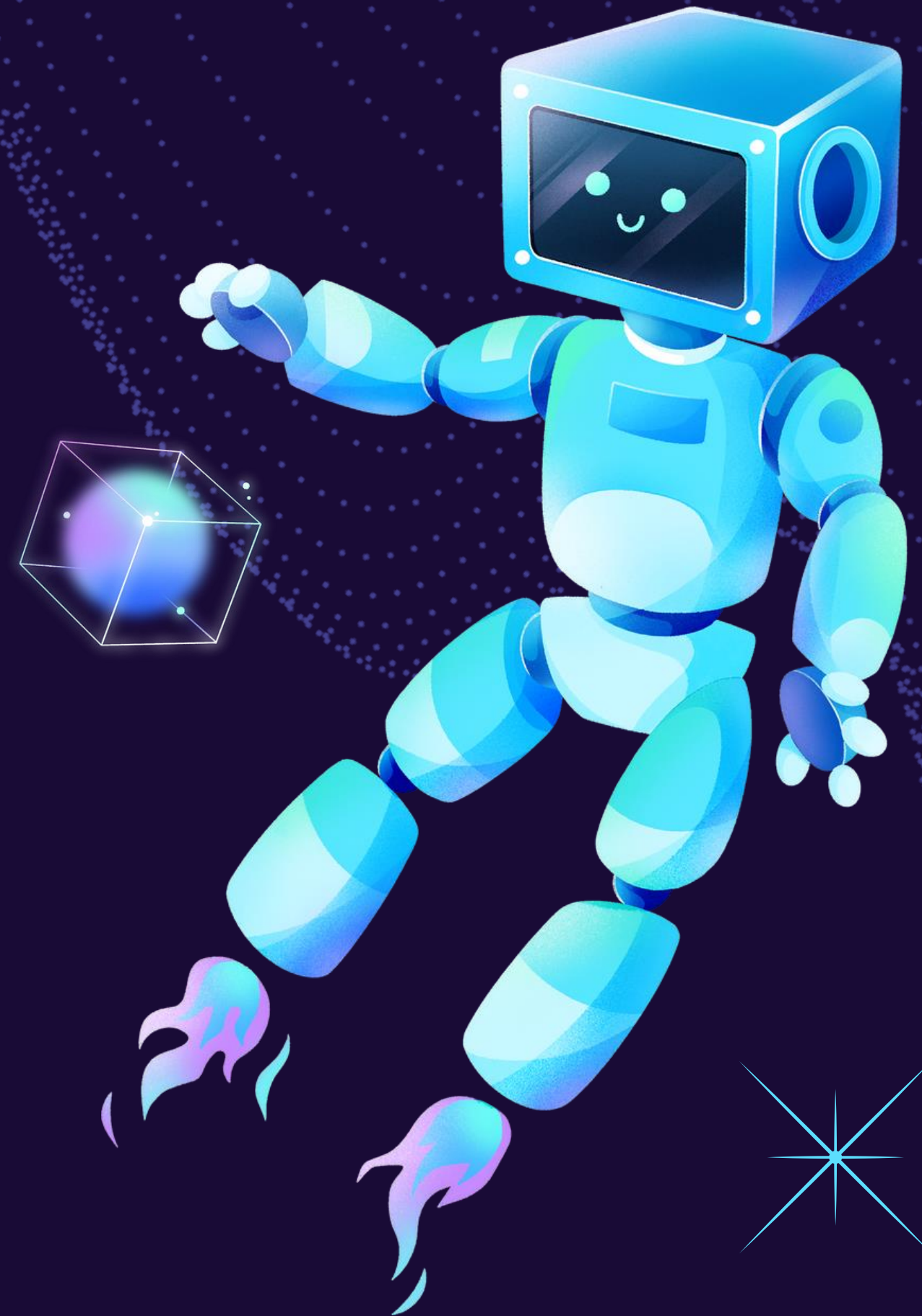
```
int main() {
    int i = 0;
    do {
        cout << i << endl;
        i++;
    } while (i < 10);
```

```
    return 0;
}
```



Hasil Run
Program:

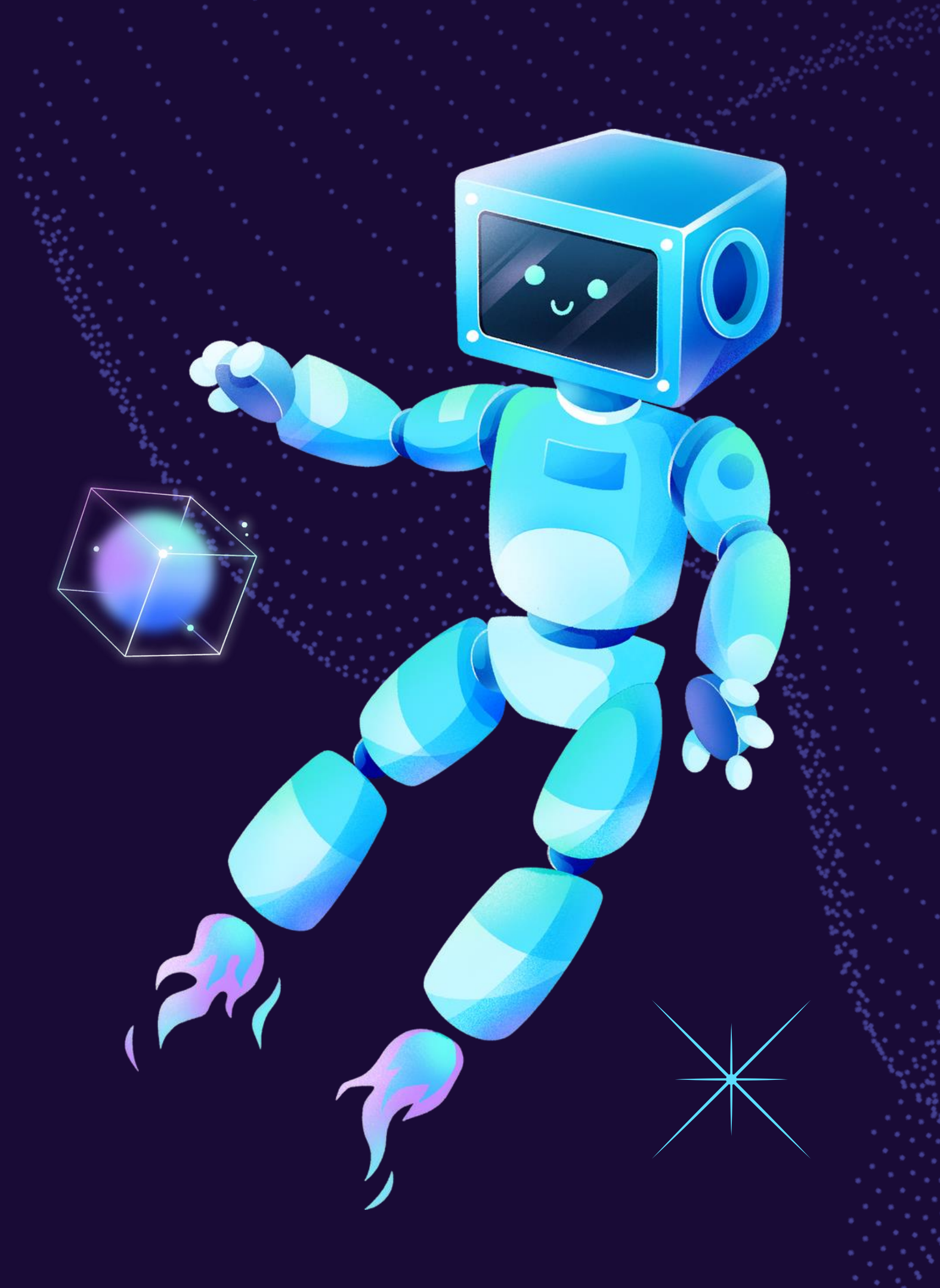
1
2
3
4
5
6
7
8
9



Pengertian array

Array adalah struktur data yang dapat menyimpan sejumlah elemen data dengan tipe yang sama dalam satu variabel. Elemen-elemen ini dapat diakses menggunakan indeks atau posisi relatif mereka dalam array. Indeks dimulai dari 0 untuk elemen pertama, 1 untuk elemen kedua, dan seterusnya.

Dengan menggunakan array, programmer dapat menyimpan dan mengelola sejumlah besar data dengan cara yang efisien. Array umumnya digunakan untuk menyimpan koleksi data homogen, seperti kumpulan bilangan bulat, bilangan desimal, karakter, atau objek dari tipe data tertentu.



a. Bagaimana cara mendeklarasikan dan menggunakan array dimensi 1 dalam

bahasa pemrograman C++?

Array adalah kumpulan variabel yang memiliki tipe data yang sama dan dinyatakan

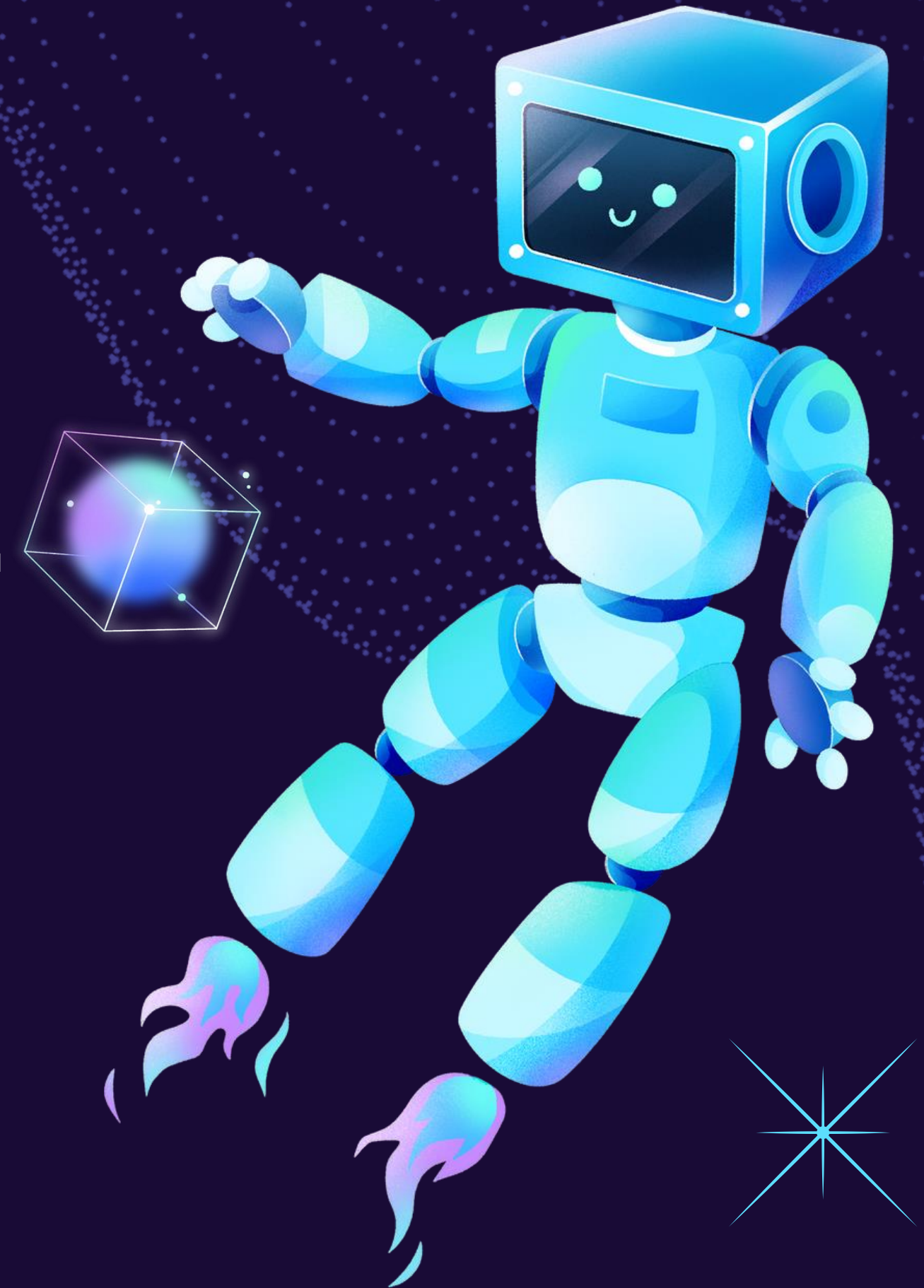
dengan nama yang sama. Array dapat berupa satu dimensi, dua dimensi, tiga dimensi

Berikut adalah langkah-langkah untuk menggunakan array satu dimensi dalam C++, ataupun banyak dimensi (multi dimensi).

Deklarasi Array: Pertama, kita perlu mendeklarasikan array yang akan digunakan. Deklarasi ini mencakup penentuan tipe data dan jumlah elemen dalam array. Contoh: `int nilai[50];`

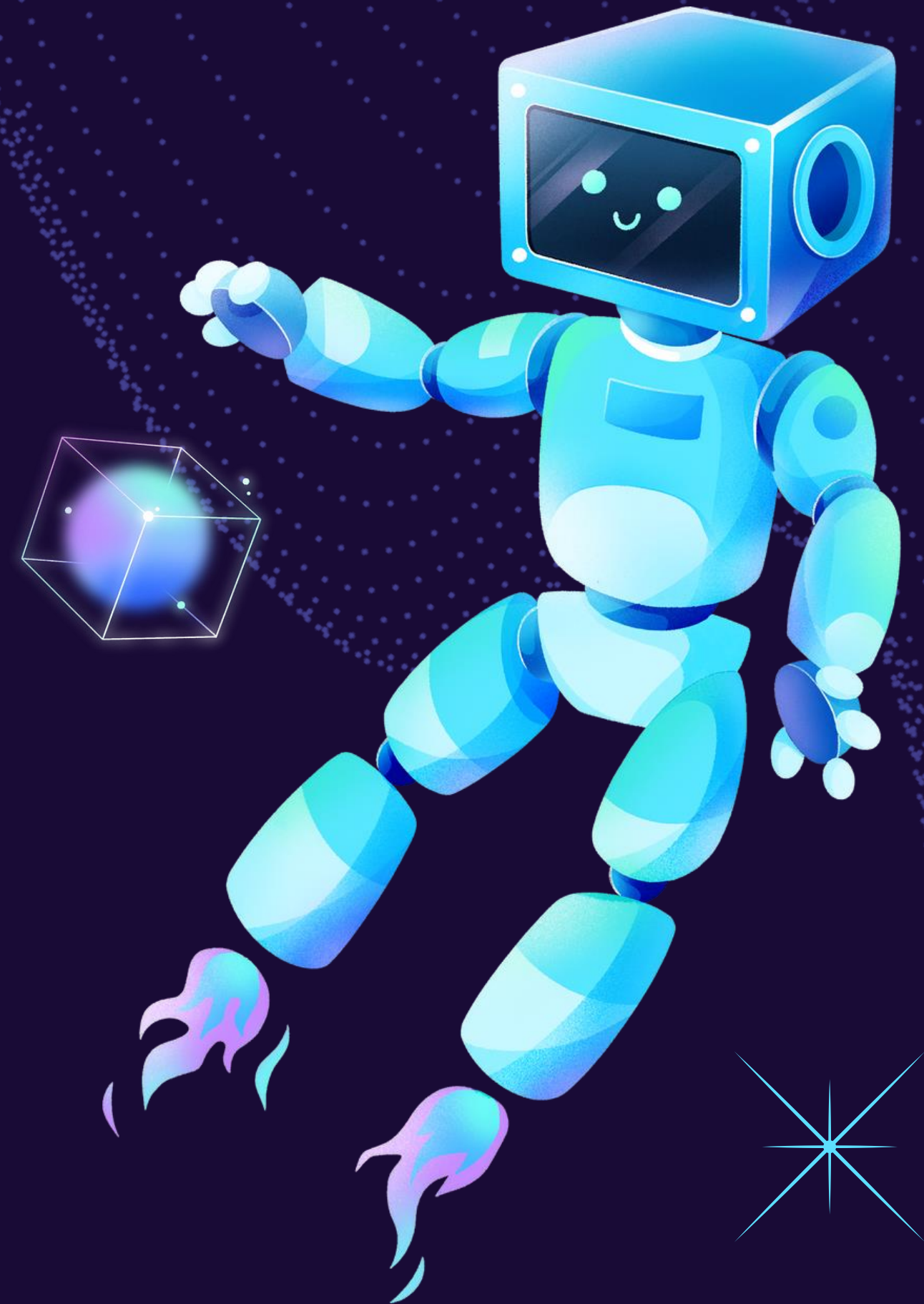
Inisialisasi Array: Selanjutnya, kita bisa memberikan nilai awal pada elemen-elemen array. Ini bisa dilakukan saat deklarasi array atau setelahnya. Contoh: `int nilai[5] = {1, 2, 3, 4, 5};`

Pengisian Array: Untuk mengisi elemen array, kita bisa menggunakan perulangan. Dengan perulangan, kita bisa mengisi elemen array secara efisien dan dinamis.



Pengaksesan Array: Untuk mengakses elemen array, kita bisa menggunakan indeks array. Indeks array dimulai dari 0.

Modifikasi Array: Untuk memodifikasi nilai elemen array, kita bisa menggunakan indeks array.



Notasi Algoritma

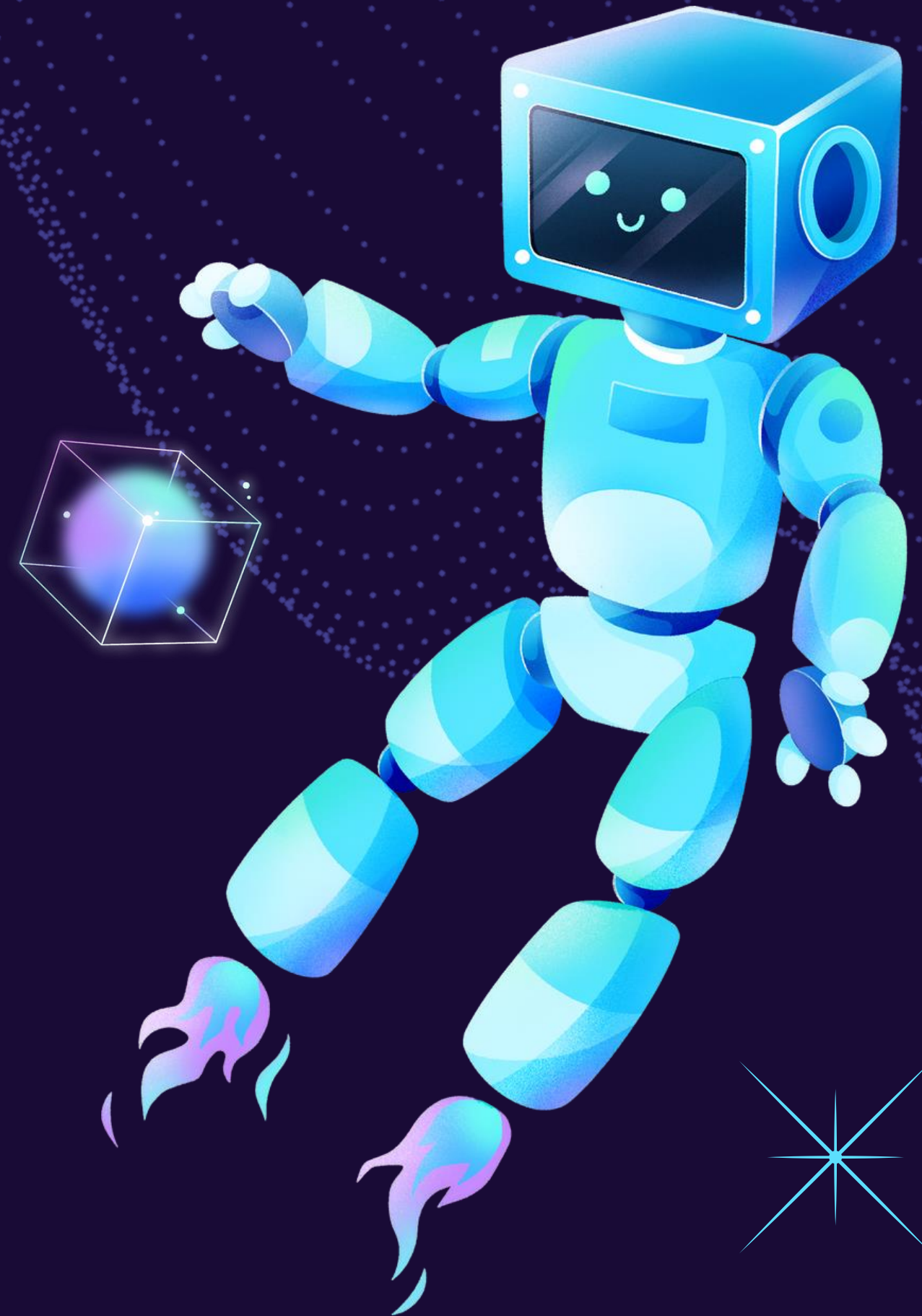
Deklarasi Array : tipe_data nama_array[jumlah_elemen];

Inisialisasi Array : tipe_data nama_array[jumlah_elemen] = {nilai1, nilai2, ..., nilaiN};

Pengisian Array : for (int a=0; a<jumlah_elemen; a++) {
 cin >> nama_array[a];
}

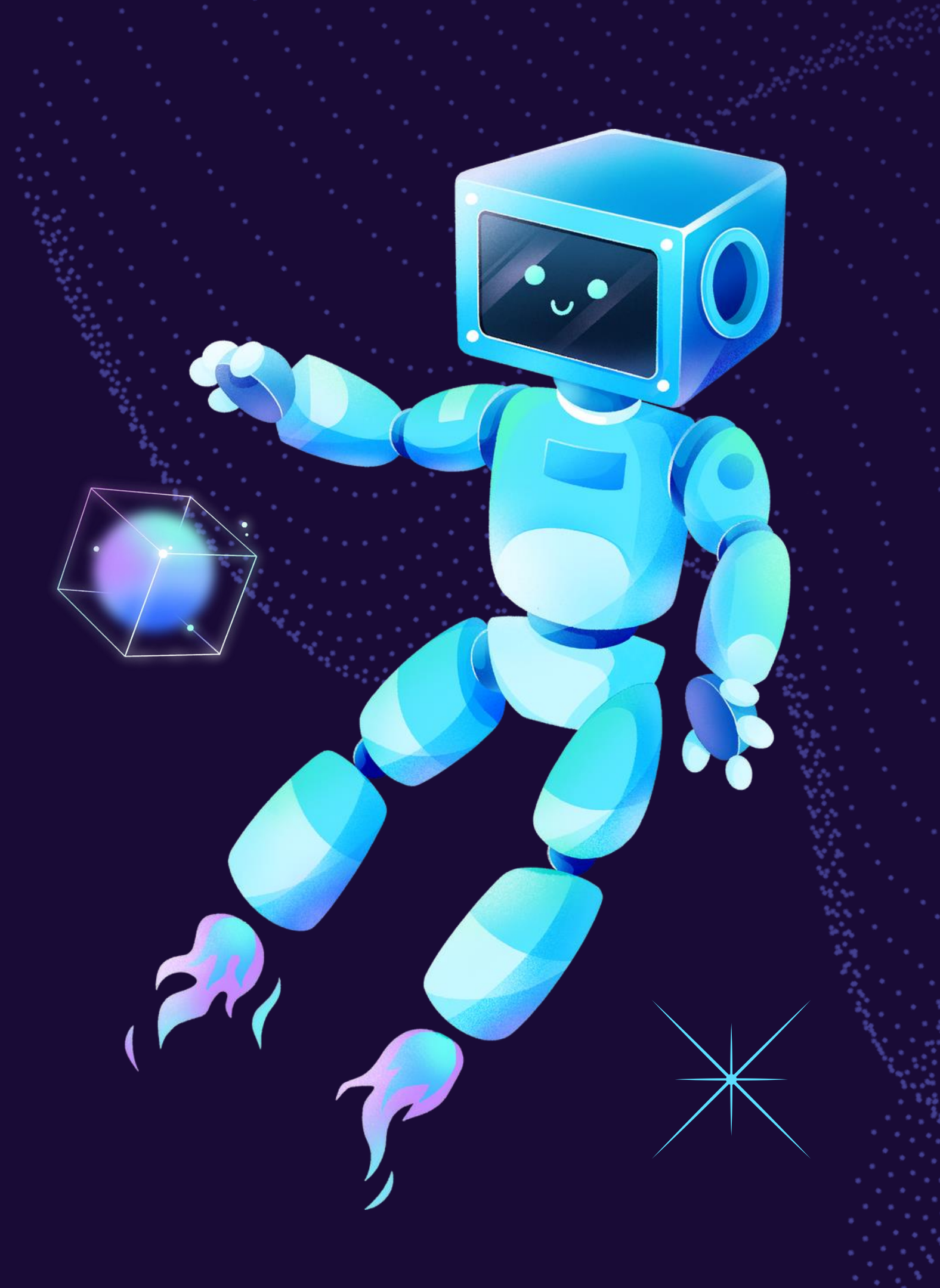
Pengaksesan Array : for (int a=0; a<jumlah_elemen; a++) {
 cout << nama_array[a] << endl;
}

Modifikasi Array : nama_array[indeks] = nilai_baru;




```
#include <iostream>
using namespace std;
int main() {
int arr[5] = {1, 2, 3, 4, 5};
  for (int i = 0; i < 5; i++) {
cout << "arr[" << i << "] = " << arr[i] << endl;
  }
return 0;
}
```

Hasil Run
Program:
arr[0] = 1
arr[1] = 2
arr[2] = 3
arr[3] = 4
arr[4] = 5



b. Bagaimana cara mendeklarasikan dan menggunakan array dimensi 2 dalam

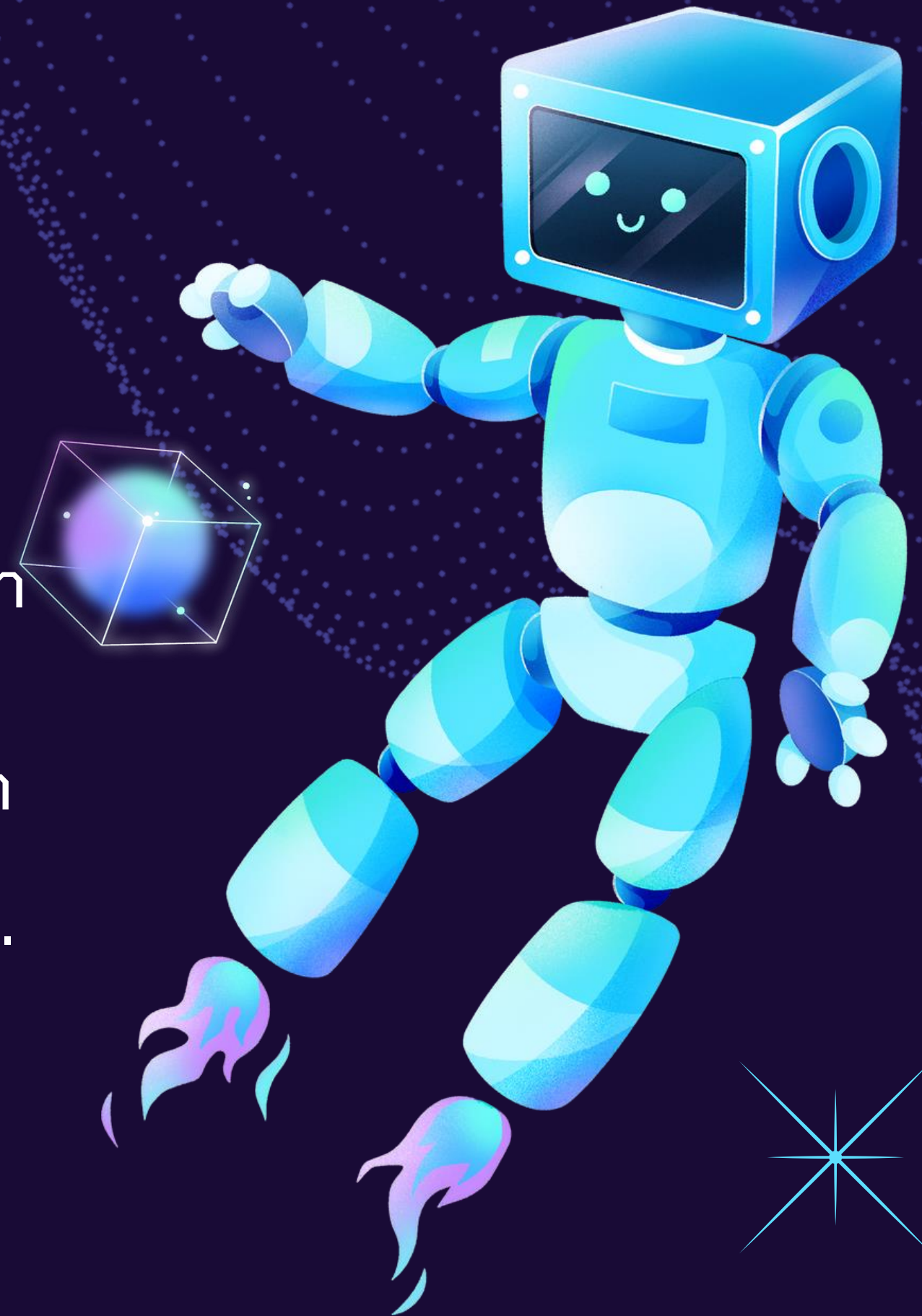
bahasa pemrograman C++?
Array Dua Dimensi

Array dua dimensi adalah sebutan untuk array yang penomorannya index-nya

menggunakan buah angka. Berikut adalah
Berikut adalah langkah-langkah untuk menggunakan
contoh penulisan array dua dimensi:
array dua dimensi dalam C++:

Deklarasi Array: Pertama, kita perlu mendeklarasikan array yang akan digunakan. Deklarasi ini mencakup penentuan tipe data dan jumlah elemen dalam array.

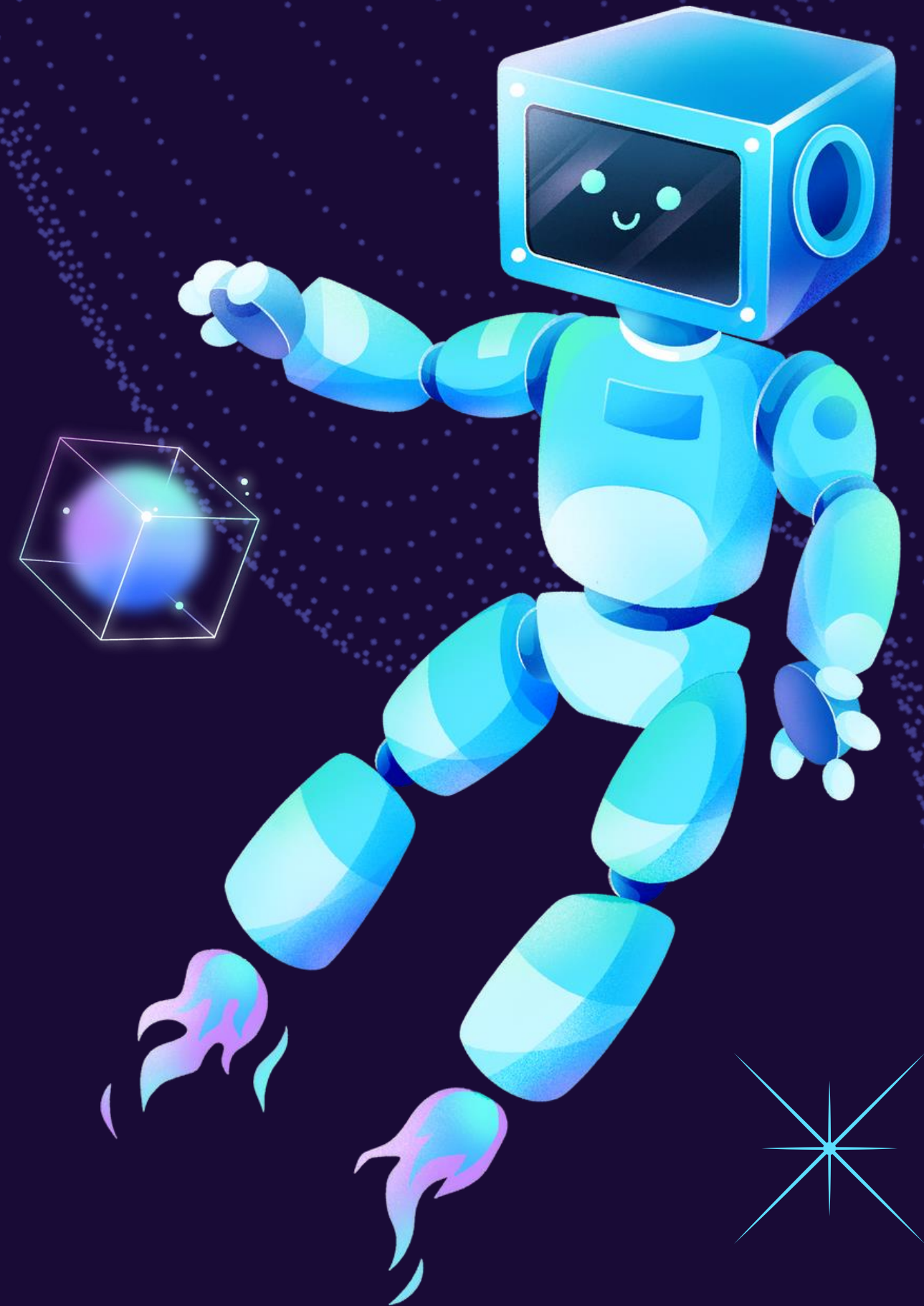
Inisialisasi Array: Selanjutnya, kita bisa memberikan nilai awal pada elemen-elemen array. Ini bisa dilakukan saat deklarasi array atau setelahnya.



Pengisian Array: Untuk mengisi elemen array, kita bisa menggunakan perulangan. Dengan perulangan, kita bisa mengisi elemen array secara efisien dan dinamis.

Pengaksesan Array: Untuk mengakses elemen array, kita bisa menggunakan indeks array. Indeks array dimulai dari 0.

Modifikasi Array: Untuk memodifikasi nilai elemen array, kita bisa menggunakan indeks array

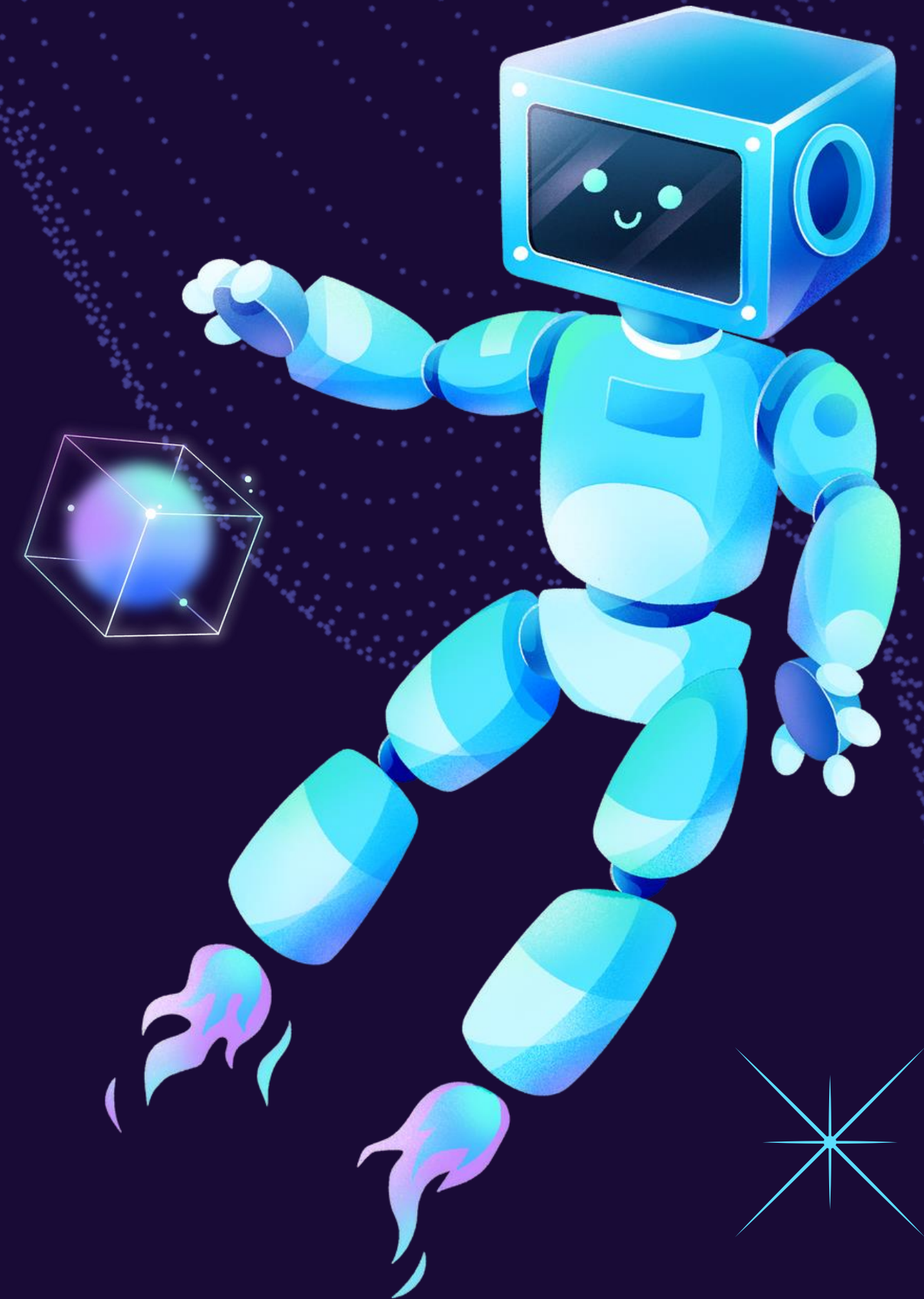


Notasi Algoritma

Deklarasi Array : tipe_data
nama_array[jumlah_elemen_baris][jumlah_elemen_kolom];

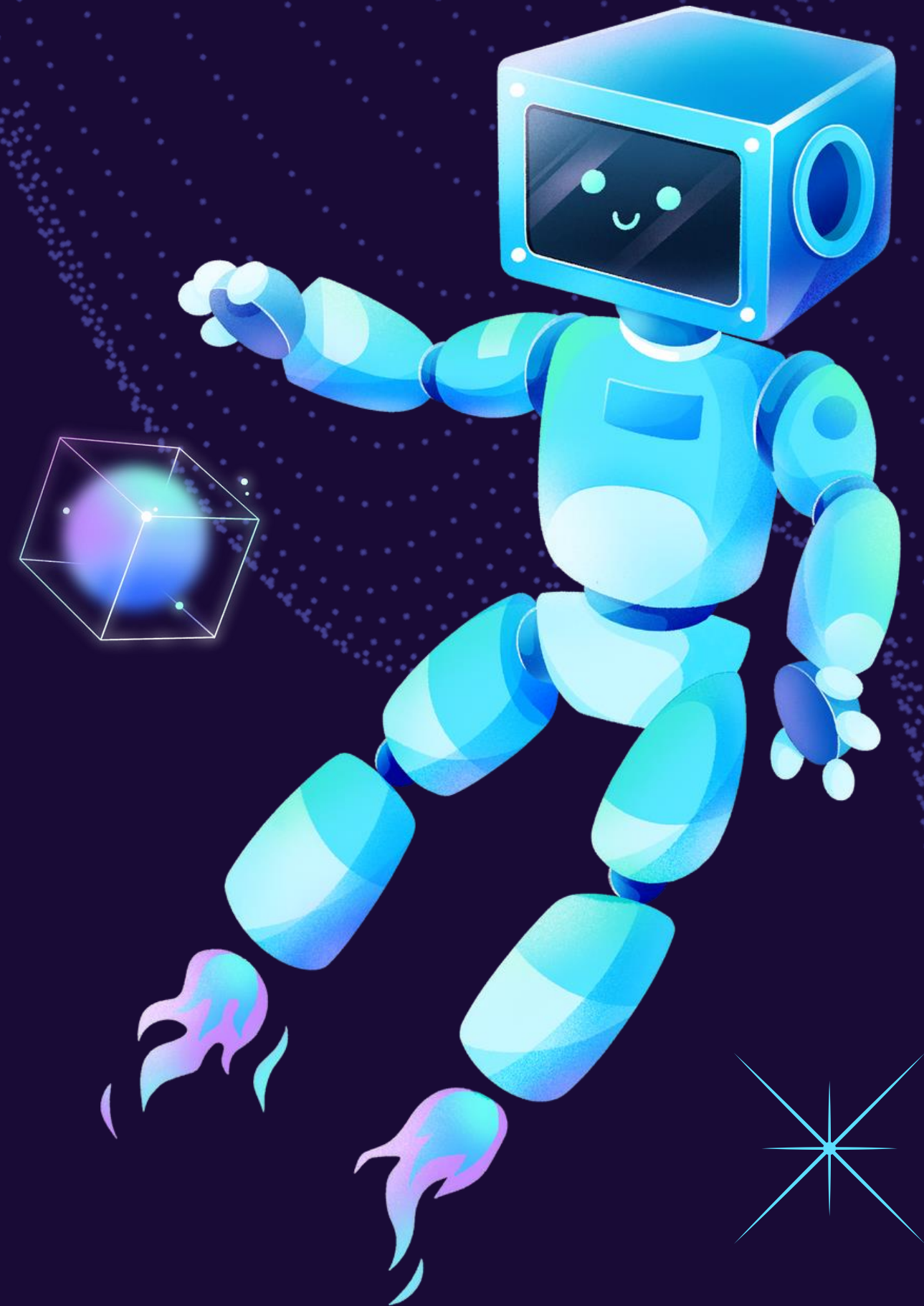
Inisialisasi Array : tipe_data
nama_array[jumlah_elemen_baris][jumlah_elemen_kolom] = {{nilai1, nilai2}, {nilai3, nilai4}};

Pengisian Array : for (int a=0;
a<jumlah_elemen_baris; a++) {
 for (int b=0; b<jumlah_elemen_kolom; b++) {
 cin >> nama_array[a][b];
 }
}




```
Pengaksesan Array : for (int a=0;
a<jumlah_elemen_baris; a++) {
    for (int b=0; b<jumlah_elemen_kolom; b++) {
        cout << nama_array[a][b] << " ";
    }
    cout << endl;
}
```

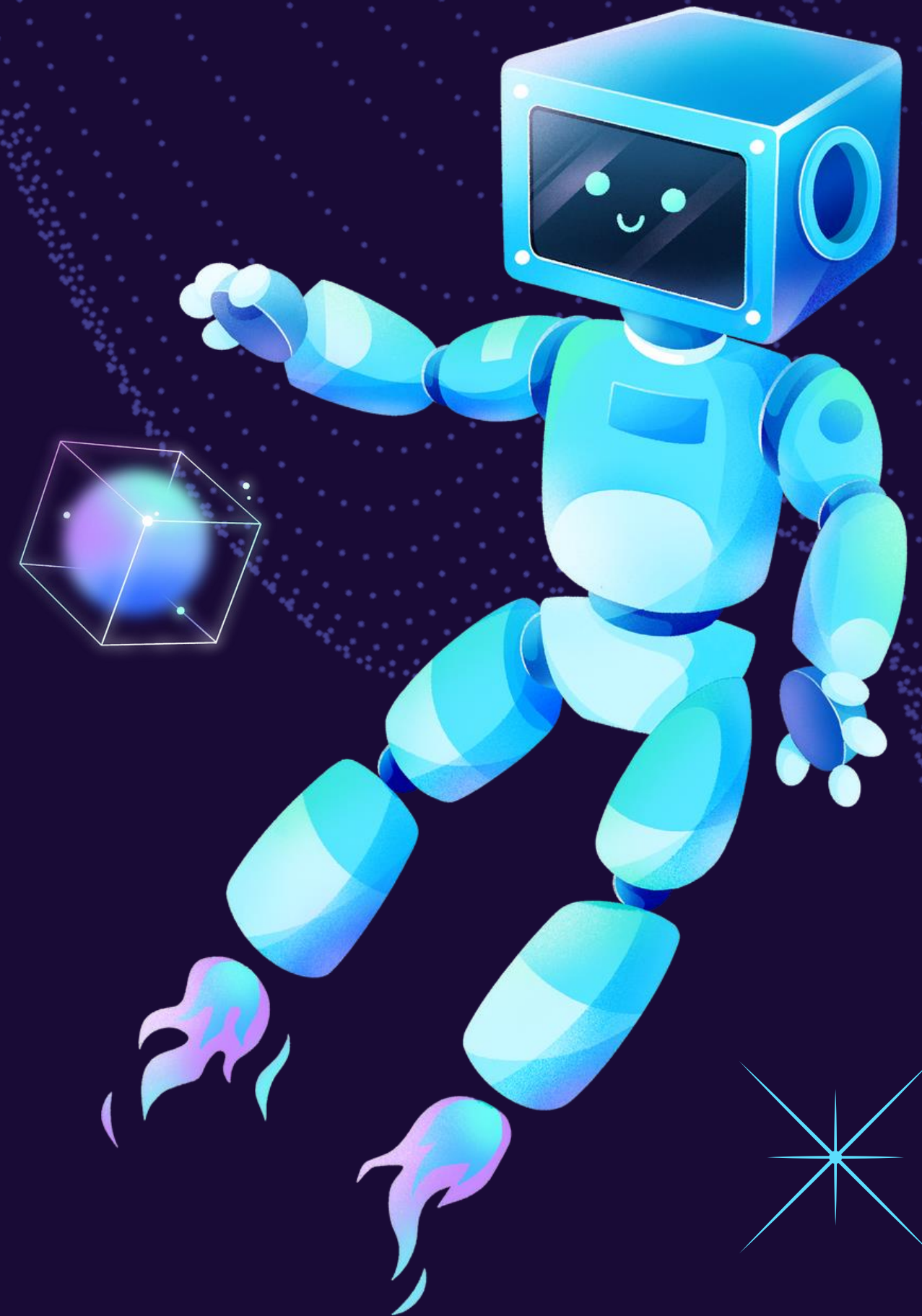
```
Modifikasi Array :
nama_array[indeks_baris][indeks_kolom] =
nilai_baru;
```



```
#include <iostream>
using namespace std;
int main() {
    int arr[2][2] = {{1, 2}, {3, 4}};
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            cout << "arr[" << i << "][" << j << "] = " << arr[i][j]
<< endl; }
        }
    return 0;
}
```

Hasil Run Program:

```
arr[0][0] = 1
arr[0][1] = 2
arr[1][0] = 3
arr[1][1] = 4
```



c. Bagaimana cara mendeklarasikan dan menggunakan array dimensi 3 dalam bahasa pemrograman C++?

Array Tiga Dimensi

Array tiga dimensi adalah array yang penomoran index-nya menggunakan 3 buah

Berikut adalah langkah-langkah untuk menggunakan array tiga dimensi. Berikut adalah contoh penulisan array tiga dimensi dalam C++:

Deklarasi Array: Pertama, kita perlu mendeklarasikan array yang akan digunakan. Deklarasi ini mencakup penentuan tipe data dan jumlah elemen dalam array. Contoh: `int arr[2][2][2];`.

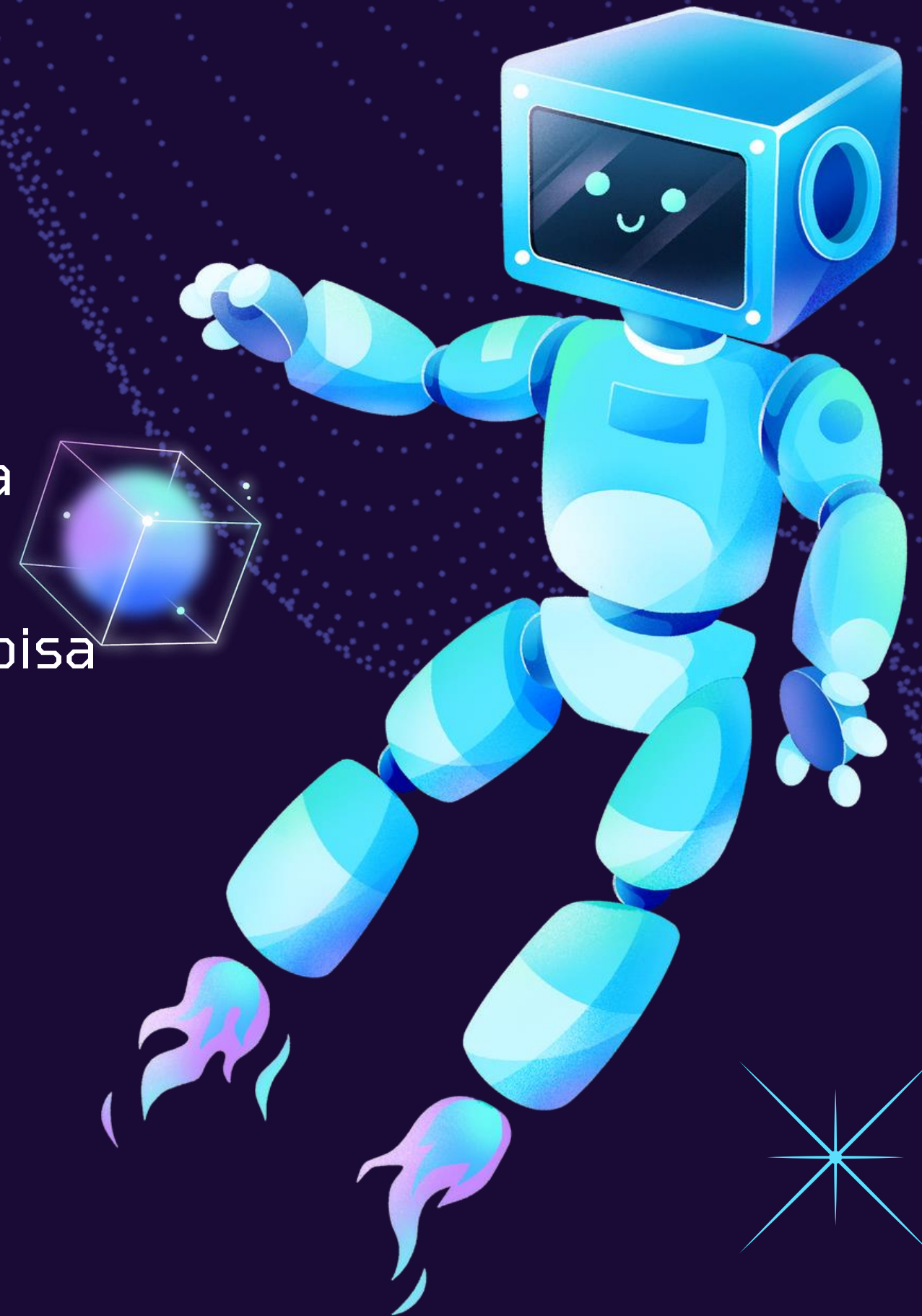
Inisialisasi Array: Selanjutnya, kita bisa memberikan nilai awal pada elemen-elemen array. Ini bisa dilakukan saat deklarasi array atau setelahnya. Contoh: `int arr[2][2][2] = {{{1,2}, {3,4}}, {{5,6}, {7,8}}};`.



Pengisian Array: Untuk mengisi elemen array, kita bisa menggunakan perulangan. Dengan perulangan, kita bisa mengisi elemen array secara efisien dan dinamis.

Pengaksesan Array: Untuk mengakses elemen array, kita bisa menggunakan indeks array. Indeks array dimulai dari 0.

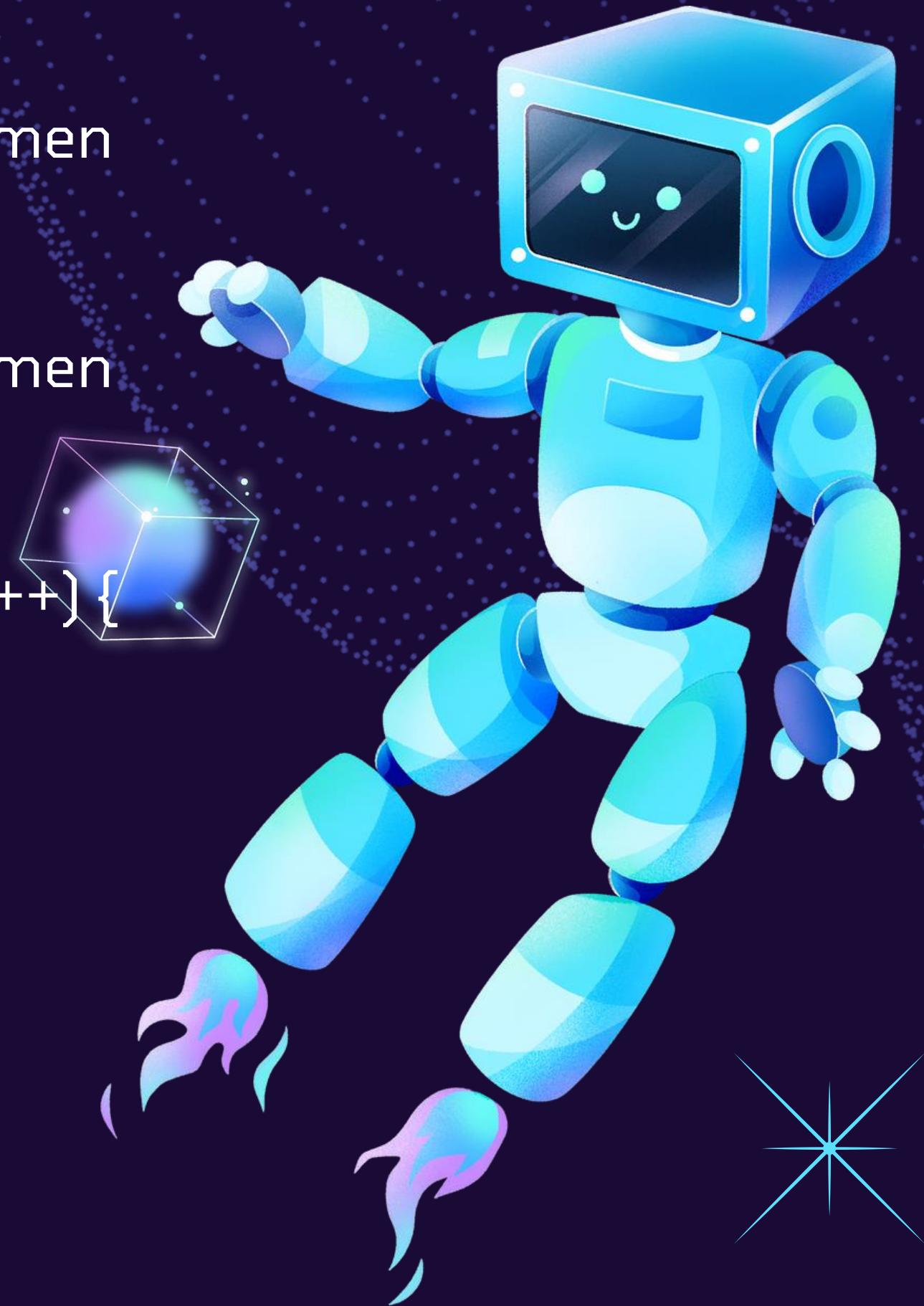
Modifikasi Array: Untuk memodifikasi nilai elemen array, kita bisa menggunakan indeks array. Contoh: `nama_array[1][1][1] = 20;`



Deklarasi Array : tipe_data
nama_array[jumlah_elemen_dimensi1][jumlah_elemen_dimensi2][jumlah_elemen_dimensi3];

Inisialisasi Array : tipe_data
nama_array[jumlah_elemen_dimensi1][jumlah_elemen_dimensi2][jumlah_elemen_dimensi3] = {{{nilai1, nilai2}, {nilai3, nilai4}}, {{nilai5, nilai6}, {nilai7, nilai8}}};

Pengisian Array : for (int a=0; a<jumlah_elemen_dimensi1; a++) {
 for (int b=0; b<jumlah_elemen_dimensi2; b++) {
 for (int c=0; c<jumlah_elemen_dimensi3; c++) {
 cin >> nama_array[a][b][c];
 }
 }
}



```
Pengaksesan Array : for (int a=0; a<jumlah_elemen_dimensi1;  
a++) {  
    for (int b=0; b<jumlah_elemen_dimensi2; b++) {  
        for (int c=0; c<jumlah_elemen_dimensi3; c++) {  
            cout << nama_array[a][b][c] << " ";  
        }  
    }  
}
```

```
Modifikasi Array :  
nama_array[indeks_dimensi1][indeks_dimensi2][indeks_dimen  
si3] = nilai_baru;
```




```
#include <iostream>
using namespace std;
```

```
int main() {
    int arr[2][2][2] = {{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}};
    for (int i = 0; i < 2; i++) {
        for (int j = 0; j < 2; j++) {
            for (int k = 0; k < 2; k++) {
                cout << "arr[" << i << "][" << j << "][" << k << "] = " << arr[i][j][k] << endl;
            }
        }
    }
}
```

```
return 0;
```

```
}
```

Hasil Run Program:

```
arr[0][0][0] = 1
```

```
12
```

```
arr[0][0][1] = 2
```

```
arr[0][1][0] = 3
```

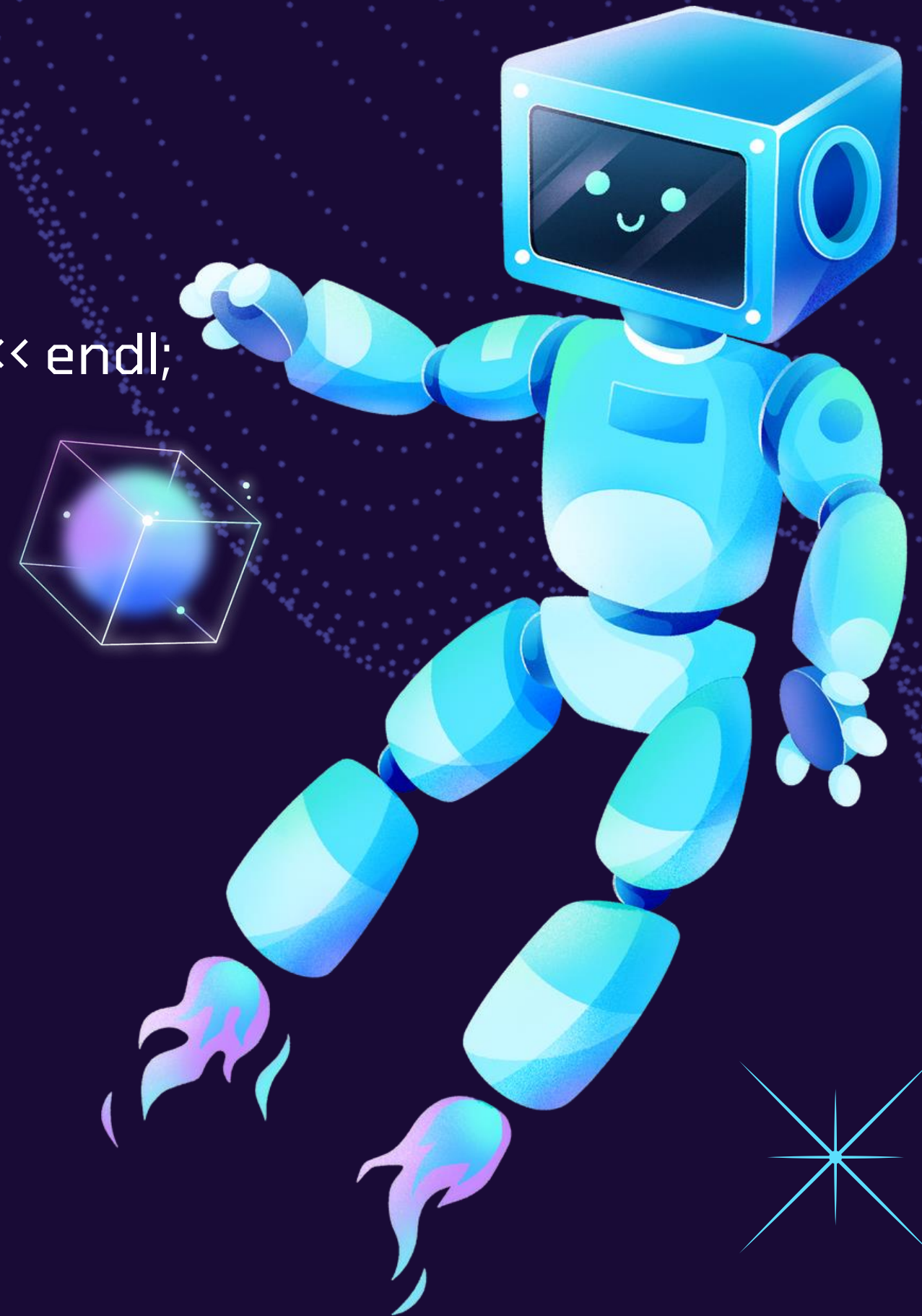
```
arr[0][1][1] = 4
```

```
arr[1][0][0] = 5
```

```
arr[1][0][1] = 6
```

```
arr[1][1][0] = 7
```

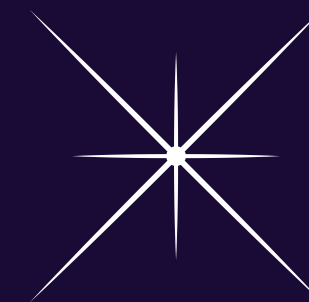
```
arr[1][1][1] = 8
```



Kesimpulan

Dalam bahasa pemrograman C++, struktur perulangan dan array memiliki peran penting dalam pembuatan program. Struktur perulangan membantu dalam penulisan kode yang efisien, sedangkan array memudahkan dalam penanganan data. Dengan memahami kedua konsep ini, programmer dapat membuat program yang lebih efisien dan efektif.





Terima
Kasih!

