

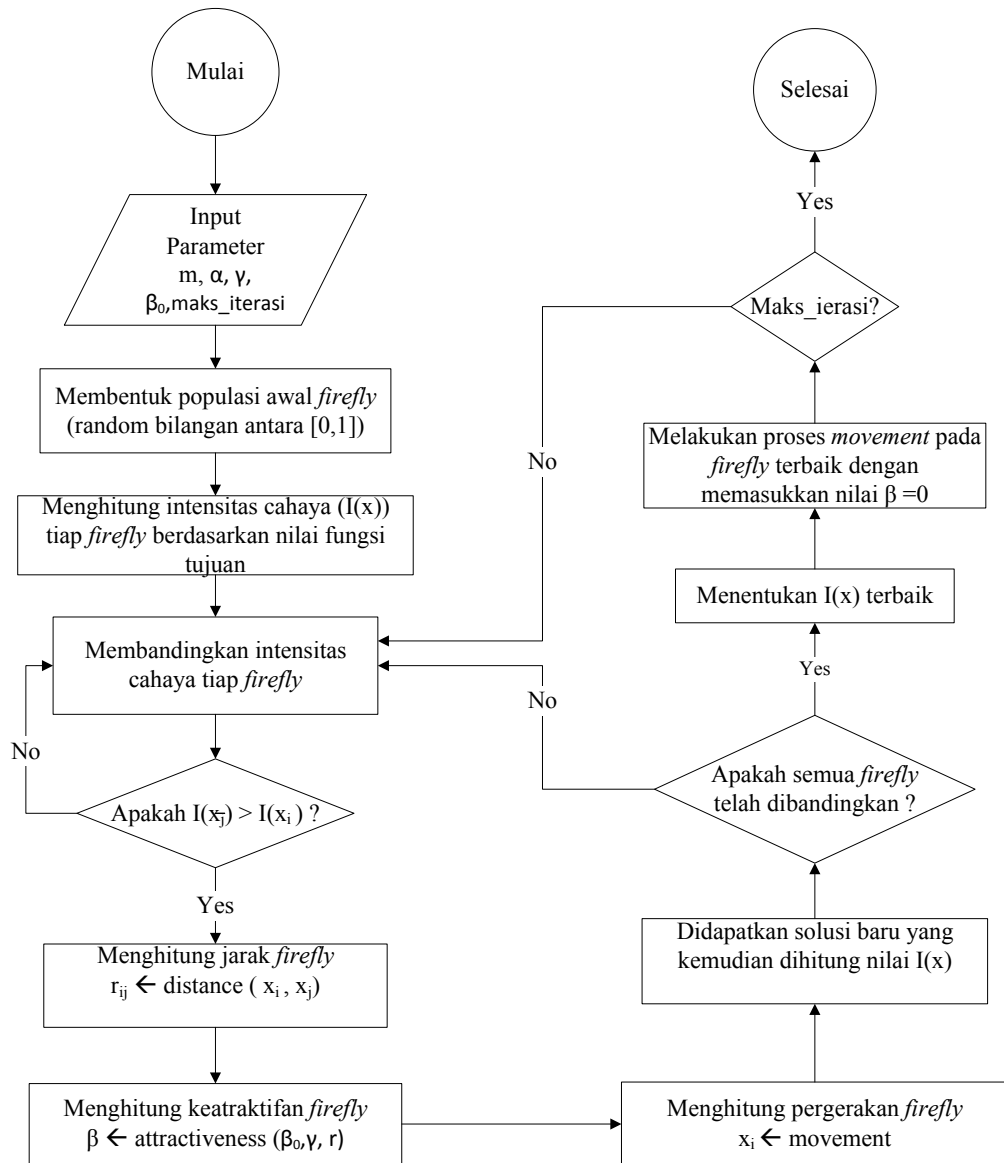
BAB III

METODE PENELITIAN

Langkah-langkah untuk penyelesaian persoalan masalah *resource constrained project scheduling* dengan *firefly algorithm* ini adalah sebagai berikut :

1. Mengkaji *resource constrained project scheduling problem* dan *firefly algorithm*.
2. Menerapkan Algoritma *Firefly* untuk menyelesaikan persoalan *Resource Constrained Project Scheduling Problem* (RCPSP), dengan langkah sebagai berikut :
 - a) Menentukan parameter-parameter dalam *Firefly Algorithm* yaitu m (banyak *Firefly*), $\alpha \in [0,1]$, $\gamma \in [0, \infty)$, $\beta_0 = 1$ dan *maks_iterasi*.
 - b) Membentuk populasi awal *firefly* dengan membangkitkan elemen-elemen yang berupa bilangan real antara $[0,1]$ secara random. Lokus dalam setiap *firefly* menyatakan nilai prioritas setiap aktivitas.
 - c) Mengevaluasi masing-masing *firefly* dengan menghitung fungsi tujuan ($f(x)$) dan Intensitas cahaya ($I(x)$). Fungsi tujuan yaitu mencari durasi tercepat dalam pengerjaan proyek. Durasi proyek dapat dihitung dengan menggunakan *Priority Scheduling Method*. Fungsi tujuan dalam permasalahan ini yaitu meminimalkan durasi sehingga intensitas cahaya masing-masing *firefly* yaitu $\frac{1}{f(x)}$.
 - d) Membandingkan intensitas cahaya tiap *firefly* dengan *firefly* lainnya.

- e) Menghitung *distance*, *attractiveness*, dan *movement* apabila menemukan *firefly* yang intensitas cahayanya lebih besar, jika tidak ditemukan *firefly* dengan intensitas cahaya yang tidak lebih besar maka akan kembali pada langkah d).
 - f) Pada saat proses update pergerakan *firefly*, didapatkan solusi baru yang kemudian dihitung intensitas cahayanya.
 - g) Cek apakah semua *firefly* telah dibandingkan. Jika semua telah dibandingkan maka lanjut ke langkah selanjutnya, jika belum dibandingkan semua maka kembali pada langkah d).
 - h) Menentukan *G-best* (solusi terbaik). *Firefly* terbaik disimpan sebagai *G-best* sementara.
 - i) Melakukan proses *movement* kepada *firefly* terbaik dan menggabungkannya dengan *firefly* yang lain untuk menjadi populasi awal pada iterasi selanjutnya.
 - j) Cek apakah sudah mencapai *max_iterasi*. Jika sudah maka selesai, jika belum akan kembali pada langkah d).
3. Membuat program dari langkah-langkah metode penelitian di atas.
 4. Mengimplementasikan program pada contoh kasus.



Gambar 3.1 Flowchart penyelesaian Resource Constrained Project Scheduling Problem menggunakan Firefly Algorithm