**ARUM YUMNA ZAHRAH**

**IFIK 40-01 / 1301164686**

**LAPORAN TUGAS KECERDASAN BUATAN**

**“MENCARI NILAI MINIMUM PADA ALGORITMA SIMULATED ANNEALING”**

1. Analisis Masalah

Diberikan suatu permasalahan matematika untuk menemukan nilai minimum dari fungsi:

dengan batasan dan .

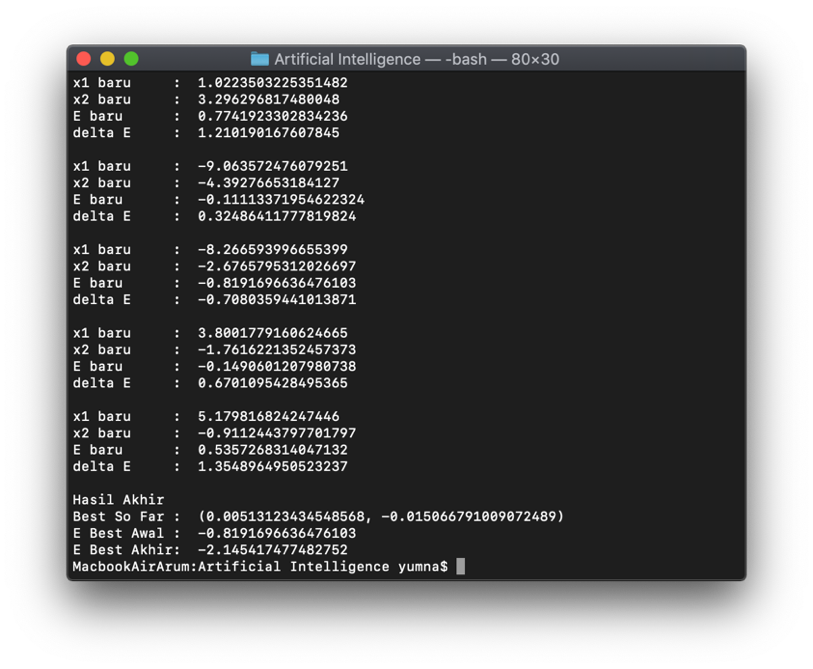
Permasalahan matematika tersebut akan diselesaikan menggunakan algoritma *Simulated Annealing* (SA) yang akan diimplementasikan ke dalam program computer menggunakan bahasa Python.

1. Strategi Penyelesaian Masalah

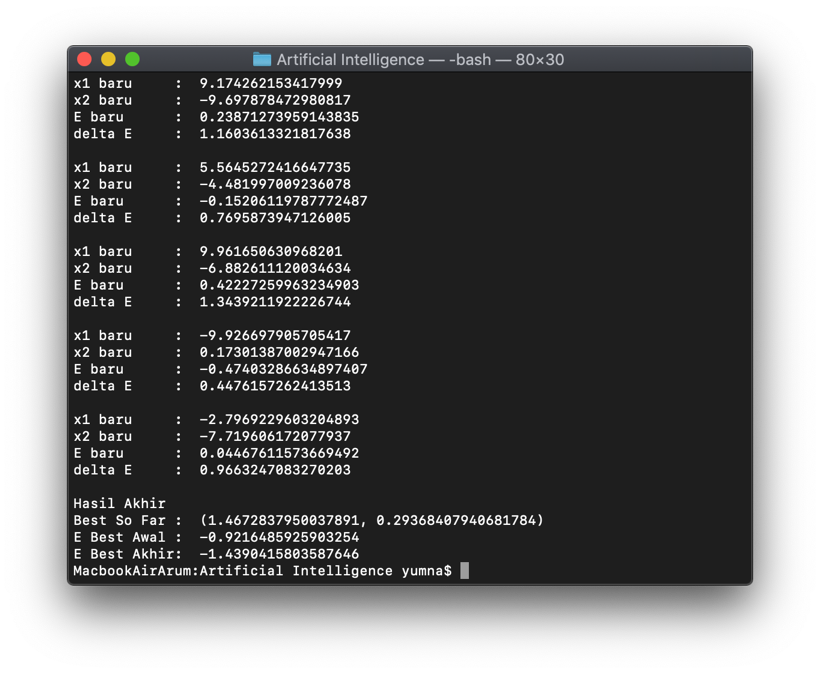
Untuk menyelesaikan permasalahan tersebut, akan dicoba 4 kali running dengan inisialisasi parameter T yang berbeda. Langkah-langkah yang digunakan adalah sebagai berikut:

1. Inisialisasikan parameter yang digunakan antara lain *x1, x2*, E0, *x1b, x2b*, Eb, deltaE, p, r, bestSofar, EBest, T, deltaT.
2. Bangkitkan bilangan *x1* dan *x2* dengan membuat function generateNewState(*x1*, *x2*) untuk mengambil bilangan random dengan interval dan .
3. Buat function E(*x1*, *x2*) untuk menghitung fungsi diatas. Hitung E0 yang merupakan hasil solusi dari fungsi diatas pada *x1* dan *x2* awal. bestSoFar diisi dengan *x1* dan *x2* awal, juga EBest diisi dengan E0.
4. Untuk running 1, akan diinisialisasi T dengan 1000 dan deltaT dengan 0.1. Untuk running 2, akan diinisialisasi T dengan 100 dan deltaT dengan 0.1. Untuk running 3, akan diinisialisasi T dengan 10 dan deltaT dengan 0.1. Dan untuk running 4, akan diinisialisasi T dengan 1 dan deltaT dengan 0.1.
5. Buat looping hingga T=0, lalu bangkitkan *x1b* dan *x2b* dengan function generateNewState(*x1*, *x2*). Kemudian hitung Eb yang merupakan hasil solusi dari fungsi pada *x1b* dan *x2b*, dan deltaE yang merupakan selisih dari Eb dan E0.
   1. Apabila deltaE kurang dari atau sama dengan 0 maka (*x1*, *x2*) diisi dengan (*x1b*, *x2b*), lalu E0 dimasukkan dengan Eb. Apabila Eb kurang dari EBest, maka bestSoFar diisi dengan (*x1*, *x2*) dan EBest diisi dengan E0.
   2. Apabila deltaE lebih dari 0 maka p diisi dengan *exp*(-deltaE/T), lalu bangkitkan r dengan bilangan random dengan interval . Apabila r kurang dari p maka (*x1*, *x2*) diisi dengan (*x1b*, *x2b*), dan E0 dimasukkan dengan Eb.
   3. Tambahkan decrement dengan T = T-deltaT.
6. Beri output hasil akhir berupa bestSoFar dan EBest. bestSoFar merupakan bilangan (*x1*, *x2*) paling optimal, sedangkan EBest merupakan nilai hasil fungsi paling minimum dan optimal.

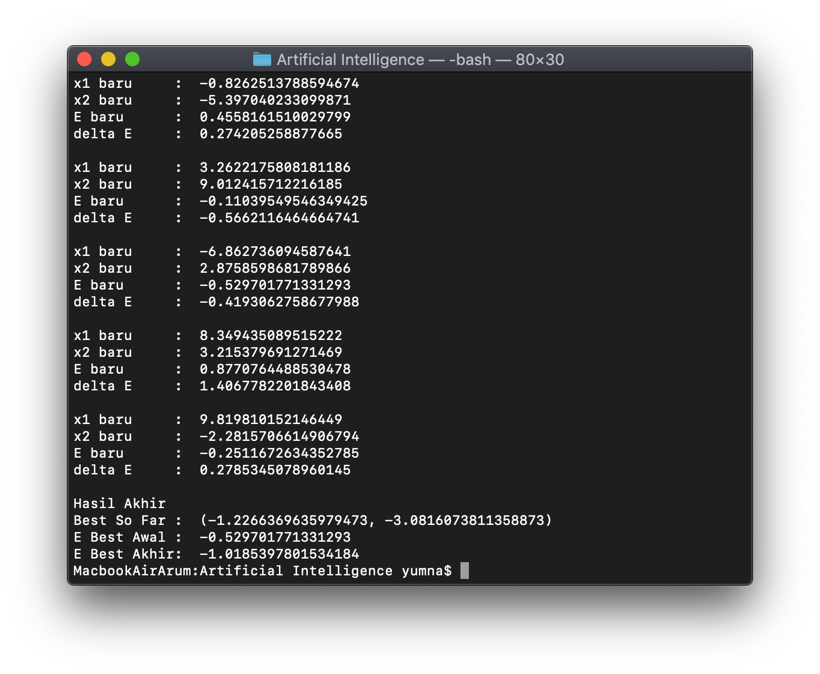
Running 1 (T=1000)



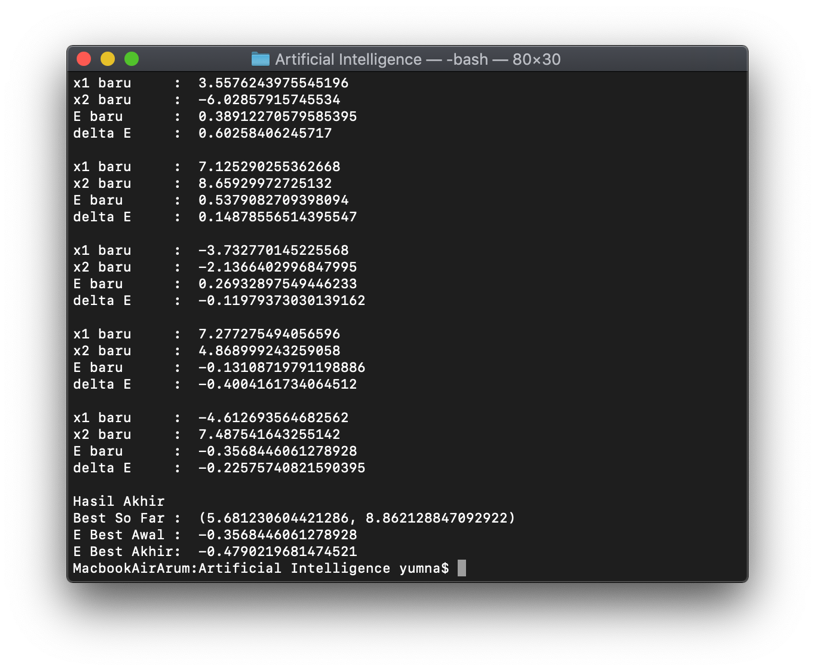
Running 2 (T=100)



Running 3 (T=10)



Running 4 (T=1)



Berdasarkan 4 kali percobaan running diatas, dapat disimpulkan bahwa semakin besar nilai parameter T, maka semakin kecil pula nilai E Best yang dihasilkan. Sehingga akan didapatkan hasil fungsi dengan nilai minimum dan paling optimal yaitu nilai E Best = -2.145 dan Best So Far dengan nilai = 0.005 dan nilai = -0.015