**Experiment No: 12**

**Aim: Write a program to minimize f(x) = x2 using Genetic Algorithm.**

**Program code:**

import random33

lO = 10

pop = [[0]\*10 for \_ in range(lO)]

npop = [[0]\*10 for \_ in range(lO)]

tpop = [[0]\*10 for \_ in range(lO)]

x = [0] \* lO

fx = [0] \* lO

m\_max = 961

ico = 0

it = 0

def iter(pp, o, p):

global m\_max, ico

sum\_val = 0

max\_val = 96

for i in range(o):

x[i] = 0

for j in range(p):

x[i] += pp[i][j] \* pow(2, p - 1 - j)

fx[i] = x[i] \* x[i]

sum\_val += fx[i]

max\_val = min(max\_val, fx[i])

avg = sum\_val / o

print("\n\nS.NO.\tPOPULATION\tX\tF(X)\n\n")

for i in range(o):

ico += 1

for j in range(p):

print(pp[i][j], end="")

print("\t\t", x[i], "\t", fx[i], "\n")

print("\nSUM: ", sum\_val, "\tAVERAGE: ", avg, "\tMINIMUM: ", max\_val)

if m\_max > max\_val:

m\_max = max\_val

def u\_rand(x):

return random.randint(0, x-1)

def tour\_sel(np, mb):

global pop, npop, fx

co = 0

while co < np:

k = u\_rand(np)

while True:

cc = 0

l = u\_rand(np)

if k != l:

break

if fx[k] > fx[l]:

npop[co] = pop[k][:]

elif fx[k] < fx[l]:

npop[co] = pop[l][:]

co += 1

cross\_ov(np, mb)

def cross\_ov(npl, mbl):

global npop, tpop

i = 0

while i < npl:

k = random.randint(0, 1)

l = u\_rand(mbl)

if k == 0 and l != 0:

npop[i][:l], npop[i+1][:l] = npop[i+1][:l], npop[i][:l]

elif k == 1 and l == mbl:

npop[i][l:], npop[i+1][l:] = npop[i+1][l:], npop[i][l:]

i += 2

for i in range(npl):

tpop[i] = npop[i][:]

mutat(npl, mbl)

def mutat(np2, mb2):

global tpop, npop, pop

for i in range(np2):

for k in range(np2):

if i != k:

r = sum(1 for a, b in zip(tpop[i], tpop[k]) if a == b)

if r != mb2 - lO:

z = u\_rand(mb2)

tpop[i][z] = 1 - tpop[i][z]

npop[k][u\_rand(mb2)] = 1 - npop[k][u\_rand(mb2)]

mutat(k, mb2)

for i in range(np2):

pop[i] = tpop[i][:]

def main():

global pop, it, m\_max, ico

n = int(input("\t\tENTER THE NUMBER OF POPULATION IN EACH ITERATION: "))

nit = int(input("\n\t\tENTER THE NUMBER OF ITERATION: "))

m = 5

for i in range(n):

for j in range(m):

pop[i][j] = u\_rand(2) # Randomize 0 or 1

print("\nITERATION ", it, " IS :\n")

iter(pop, n, m)

it += 1

do\_continue = True

while do\_continue and it < nit:

it += 1

print("\nITERATION ", it, " IS :\n")

tour\_sel(n, m)

iter(pop, n, m)

do\_continue = it < nit

print("\n\nAFTER THE ", ico, " ITERATION, THE MINIMUM VALUE IS ", int(m\_max \*\* 0.5))

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Output:**



