# -\*- coding: utf-8 -\*-

"""2\_19101652\_Muhtasim Mahmud\_Assignment2.ipynb

Automatically generated by Colaboratory.

Original file is located at

https://colab.research.google.com/drive/12WT8DufhZ9Faclos3s3\_EL1gmiLIbPbE

"""

import numpy as py

import random

plus\_inf = +9999999

minus\_inf = -9999999

#position is current index

def alpha\_beta(position, alpha, beta, distance, maximizing, child\_nodes):

global depth

if distance == b:

return child\_nodes[position]

if maximizing:

O\_value = -9999999

for i in range(0, b):

value = alpha\_beta(position\*a + i, alpha, beta, distance+1, False, child\_nodes)

O\_value = max(O\_value, value)

alpha = max(O\_value, alpha)

if alpha >= beta:

depth-=1

break

return O\_value

else:

O\_value = plus\_inf

for i in range(0, a):

value = alpha\_beta(position\*a + i, alpha, beta, distance+1, True, child\_nodes)

O\_value = min(O\_value, value)

beta - min(O\_value, beta)

if alpha >= beta:

depth-=1

break

return O\_value

# main start from here

student\_id = "17301106" # i have taken this id because of id[2] = 3 which makes this more clear

b = int(student\_id[0])\*2 #how many times attacker can attack

initial\_hp = student\_id[6::] #last two digits of id

initial\_hp = initial\_hp[::-1] #reversing the last two digits and this is initain lifeline

a = int(student\_id[2]) #attacker will take dcsn from how many child

depth = 0

total\_nodes = a \*\* b

negative\_HP = input("negative HP range :")

negative\_HP = negative\_HP.split(" ") #doing split of the user given range

minimum\_negative\_HP = int(negative\_HP[0])

maximum\_negative\_HP = int(negative\_HP[1])

print("1. Depth and Branches ratio is : "+ str(b) + ":" + str(a))

child\_nodes = [random.randint(minimum\_negative\_HP, maximum\_negative\_HP) for i in range(0, total\_nodes)]

print("2. Terminal States(leaf node values) are ", \*child\_nodes, " ")

current\_hp = alpha\_beta(0, minus\_inf, plus\_inf, 0, True, child\_nodes)

remaining\_hp = int(initial\_hp) - int(current\_hp)

print("3. Left life(HP) of the defender after maximum damage caused by the attacker is", remaining\_hp)

depth = len(child\_nodes)

x = alpha\_beta(0, minus\_inf, plus\_inf, 0, False, child\_nodes)

print("4. After Alpha-Beta Pruning Leaf Node Comparisons", depth)