#Brooks Pettit

#Hannah Schroeder

#Conner Brewster

#Import various libraries

import random

import time

#Global variables

health = 100

inventory = ['flashlight']

level = 0

win = False

#define all functions

def introduction():

enter = False

#Game instructions

print("Welcome to the Great Pyramid Treasure Hunt.\n\n",

"You are a world renowned treasure hunter who has traveled\n",

"thousands of miles to this mysterious and exotic location\n",

"in search of riches and glory.\n\n",

"To control your character, type simple commands into the prompt.\n",

"The computer will only understand simple commands, so try\n",

"to be as brief as possible. For example, instead of typing\n",

"'LOOK AT THE DOOR', you might try 'INSPECT DOOR', and instead\n",

"of 'PICK UP THE ROCK AND PUT IN POCKET', you might try 'TAKE ROCK'\n"

"Try 'LOOK'ing around for a start.")

print("\nFor a list of some generic commands, type 'help' before\n",

"you enter the pyramid and begin your adventure.\n")

while not enter:

#Get command

action = input("\n> ")

action = action.lower()

#Possible commands

if action == 'inventory':

for element in inventory:

print(" -", element)

#Help

elif action == 'help':

print("If you are stuck, try the following commands:")

print("\nLOOK\nOPEN\nINSPECT\nREAD\nUSE\nGIVE\nTAKE",

"\nSome of these will requre an object in conjunction with the command",

"\n\nYou may be able to do other simple things in",

"certain situations")

elif action == 'health':

print(health)

elif action == 'look':

print("\nIn front of you looms a gigantic, ancient pyramid.\n",

"A huge set of double doors dominates the face of the\n",

"pyramid. Somthing appears to be carved above the doors.")

elif action == 'inspect pyramid':

print("\nThe pyramid is made of smooth stone. Apart from it's\n",

"monstorous size and large entrance, there is nothing\n",

"particularly interesting about it")

elif action in ('inspect doors', 'inspect door', 'inspect entrance'):

print("\nTaking a closer look, you notice that there is no\n",

"lock or other barrier on the doors. It is now clear\n",

"that the carvings above are a poem of some sort.")

elif action in ('inspect carving','inspect carvings','inspect poem','read poem'):

print("\nThe carving is in a language you do not recognize.\n",

"After staring at the characters for some time,\n",

"they morph into English")

print("It reads")

print("\tShould you fail to pass the test,")

print("\tYou'll die within like all the rest.")

elif action in ('open door','enter pyramid','open doors'):

enter = True

print("\nYou give the door a tug. It's a rather heavy door. But\n",

"it swings open easily enough.")

##############################################################################################################

def level1(health, inventory):

print("You step into a huge room. The door swings shut behind you.")

#local variables

exit\_room = False

door = 'locked'

sarcophagus = ['medallion', 'key', 'closed']

while not exit\_room:

action = input("\n> ")

action = action.lower()

if action == 'inventory':

for element in inventory:

print(" -", element)

#The cheats are so we can get to later levels fast

elif action == 'cheat':

exit\_room=True

#List all possible actions

elif action == 'health':

print(health)

if action == 'look':

print("\nThe room you are in is very large and bare.\n",

"Directly in front of you there is another door,\n",

"much smaller than the first. To your left, there is\n",

"a long sarcophagus.")

elif action == 'inspect door':

print("\nNext to the door, a skeleton hand protrudes from the wall.\n",

"Above the hand, you see the enscription:\n",

"\t\t'Tribute'")

elif action == 'open door':

if door == 'locked':

print("\nThe door is locked.")

elif door == 'unlocked':

door = 'open'

print("\nThe door swings open easily, revealing a staircase leading up.")

else:

print("Door opened.")

door = 'open'

elif action == 'inspect hand':

print("\nThere is nothing particularly interesting about\n",

"the hand. It looks as if it is waiting for somthing")

elif action == 'inspect sarcophagus':

print("\nThe sarcophasgus is very elaborate. It is plated\n",

"in gold and jewels. On the lid, the words 'A theif's reward'\n",

"appear.")

elif action in ('open sarcophagus','lift lid'):

if 'closed' in sarcophagus:

sarcophagus.remove('closed')

if 'medallion' in sarcophagus and 'key' in sarcophagus:

print("\nThe lid creaks open. Inside there is an ancient\n",

"mummy. Arround the mummy's neck there is a\n",

"gold medallion. A small metal key lies next to the mummy")

elif 'medallion' in sarcophagus:

print("\nThe lid creaks open. Inside there is an ancient\n",

"mummy. Arround the mummy's neck there is a\n",

"gold medallion.")

elif 'key' in sarcophagus:

print("\nThe lid creaks open. Inside there is an ancient\n",

"mummy. A small metal key lies next to the mummy")

else:

print("\nThe lid creaks open. Inside there is an ancient\n",

"mummy.")

elif action == 'take medallion':

if 'medallion' in sarcophagus and 'closed' not in sarcophagus:

sarcophagus.remove('medallion')

inventory.append('medallion')

print("\nYou have added the medallion to your inventory.")

elif 'medallion' in inventory:

print("\nYou already have this item.")

elif action == 'take key':

if 'key' in sarcophagus and 'closed' not in sarcophagus:

sarcophagus.remove('key')

inventory.append('key')

print("\nYou have added the key to your inventory.")

elif 'medallion' in inventory:

print("You already have this item.")

elif action in ('give medallion', 'place medallion'):

if 'medallion' in inventory:

door = 'unlocked'

print("\nYou cautiously drop the medallion into the skeleton's hand.\n",

"The bony fingers crack as they close arround the gold. You\n",

"hear a light click behind the door")

inventory.remove('medallion')

elif action in ('give key', 'place key'):

if 'key' in inventory:

print("\nNothing happens. You slide the key back into your pocket")

elif action == 'shake hand':

health -= 10

print("\nYou clasp the skeleton's hand. Suddenly its grip hardens\n",

"and you cannot break free. You struggle against the hand.\n",

"It loosens its grip and you fall backwards, striking your\n",

"head on the stone floor.\n(health - 10)")

elif action in ('climb stairs', 'climb staircase', 'enter door'):

if door == 'open':

#Next level

print("\nYou ascend the stairs to the next level.")

exit\_room = True

elif door == 'unlocked':

print("\nThe door is still closed.")

else:

print("\nThe door is locked.")

if health<0:

#Game over

exit\_room = True

return health,inventory

#######################################################################################################################

def level2(health, inventory):

#local variables

print("You enter a slightly smaller room.")

exit\_room = False

no\_mirror = False

no\_lock = False

no\_bush = False

eyes = 2

while exit\_room == False:

#choice is the same as action

choice = input("\n> ")

choice = choice.lower()

if choice == "nope":

print("You decide that you are a rabid monkey. You charge head first into the wall.\nNot surprisingly, it makes a loud clank and you fall in pain.")

health -= 25

#The cheats are so we can get to later levels fast

elif choice == 'cheat':

exit\_room=True

#All avaialble commands

elif choice == "inventory":

for element in inventory:

print(" -", element)

elif choice == "explode":

print("You blew up.")

break

elif choice in ["throw flashlight", "hurl flashlight", "fling flashlight"]:

print("You blindly hurl your flashlight upwards without looking.\nIt merely lands on your head with a thud. No blood clot occurs, but it hurts.\nYou wonder how you made it this far in life while\nmaking decisions similar to that.\nYou place the flashlight back in your pockets.")

health -= 5

elif choice in ["throw key", "hurl key", "fling key"]:

print("You blindly hurl your key upwards without looking.\nIt merely lands in your eye. You now cannot see out of that eye.\nSeriously?\nYou place the key back in your pockets.")

eyes -= 1

health -= 5

if eyes == 0:

print("\nYou are now completely blind.\nDidn't you learn from the first time you threw your keys that\nit was a bad idea?")

break

elif choice in ["throw mirror", "hurl mirror", "fling mirror"]:

if "Mirror" in inventory:

print("You blindly hurl your mirror upwards without looking.\nIt slams into your skull and shatters. It hurts, and the mirror is completely destroyed.")

del inventory [2]

health -= 10

elif choice in ["look at flashlight", "inspect flashlight"]:

print("You take a closer look of the flashlight.\nTurns out... it's a flashlight.\nGo figure.")

elif choice == "health":

print(health)

elif choice == "take pyramid":

print("You clutch the ground beneth you and stuff the\npyramid in your pocket.\nHowever, since you are inside of the pyramid, you also go in your pockets,\nHowever, since your pyramid is in your pocket of you who is inside the pyramid and\naiwdhapIDHiahdipuahdpihdpahdiphdpaodjoadjpoajpdjpjdiebhfouehfehfuioefhaeufh08wq34hy0897523y90823ynv08732nf89y3v07283vb87vn327cf897c7829379`0010101010100101-0101010101001010101001010100101010100101")

break

elif choice == "open flashlight":

if "screwdriver" in inventory:

print("Impossible.")

break

else:

print("You are unable to open the flashlight,\na proper tool is required.")

elif choice == "look":

print("\nYou find a dusty mirror hanging angled on the left wall,\nsome hieroglyphics to the right,\na red box left of that,\nand dense thorny bushes.")

elif choice in ["look in mirror", "see mirror", "examine mirror", "inspect mirror"]:

if no\_mirror == False:

print("Inside the relection you see yourself, looking more sexy than ever.\n...and a dark object approaching you at an incredible rate.\nBefore you could react, you are whisked away.")

print("\nBecause the creature doesn't have a degree in\ntransportation safety, you are hurt.")

health -= 10

else:

print("For some reason the mirror cannot be seen inside anymore.\nIt just looks like a slab of bronze.")

elif choice in ["take mirror", "acquire mirror", "steal mirror", "grab mirror"]:

if no\_mirror == False:

print("Without looking inside you detach the mirror\nfrom the wall and put it in your pocket...\nhow the heck did you do that?")

inventory += ["Mirror"]

no\_mirror = True

print("\nBehind where the mirror was lies some text.\nOddly enough, it's in English lettering.")

print("\nIt reads: htiw detrats yeht tahw esol tsum eno ,htrae detaeh eht nihtiw")

else:

print("\nYou look at the text once more...")

print("\nIt reads: htiw detrats yeht tahw esol tsum eno ,htrae detaeh eht nihtiw")

elif choice in ["look at hieroglyphics", "inspect hieroglyphics"]:

print("You translate the text...\n\nIT.. IT DON GIV MERCI NO LOOKIE\n\nYou then wonder why the text looks smeared in\na bloody red...")

elif choice in ["take box", "acquire box", "steal box", "grab box"]:

print("You grab the box and lift it up\nspikes pop out from the sides of the box and pierce you.\nIn instinct, you drop the box...\nright on top of your foot.\nYour cries of agony echo loudly throughout the room.")

health -= 20

elif choice in ["inspect box"]:

print("You take a closer look at the red box...\nIt seems out of place, a little modern for an ancient pyramid.")

if no\_lock == False:

print("You see that it is tightly shut with a key lock in place.")

else:

print("You see the place where the Flint and Steel once rested.")

elif choice in ('use key','unlock box','unlock lock','open box'):

if 'key' in inventory:

print("With a quick click, the red box's lid pops off.\nWithin you find a Flint and Steel. You grab it.")

no\_lock = True

inventory += ["Flint and Steel"]

else:

print("You don't have a key.")

elif choice in ["look at thorns", "inspect thorns"]:

print("You step closer to the dense bushes.\nYou feel nervous around their incrediblly sharp thorns.\nAt the end you see a light shimmering from inside.")

elif choice in ["enter thorns", "go in thorns", 'enter bush','enter bushes']:

if no\_bush == True:

print("You easily slide through the bushes.\n\nYou find yourself in front of a ladder. You climb it.")

break

else:

print("You attempt to push through the bushes,\nbut you are severely wounded from the sharp thorns.\nYou retreat back for now into the main area.")

health -= 10

elif choice in ["ignite flint and steel", "use flint and steel"]:

if "Flint and Steel" in inventory:

print("You immediately trigger your Flint and Steel\nwhich happens to be in your pocket.\nAn intense burning follows, resulting in singed pants.")

health -= 25

else:

print("What Flint and Steel?")

elif choice in ["burn thorns", "burn bush", "burn bushes"]:

if "Flint and Steel" in inventory:

print("You go near the bushes and light the Flint and Steel.\nThe thorns disappear without a trace.")

no\_bush = True

else:

print("What Flint and Steel?")

else:

print("You scream", choice,"hoping that it is some sort of password.")

#Is the user alive?

if health<0:

exit\_room = True

return health,inventory

#This is a function for displaying a puzzle that appears later in level 2

#Every time the user does something, the puzzle displays

def display\_puzzle(letters):

print(" \_\_\_ \_\_\_ \_\_\_ \_\_\_ ")

print("| | | | |")

print("|", end=" ")

for i in range(2):

for i in range(i\*4,(i\*4)+4):

print(letters[i],"|", end=" ")

print("\n|\_\_\_|\_\_\_|\_\_\_|\_\_\_|")

print("| | | | |")

print("| ", end="")

for i in range(8,12):

print(letters[i],"|", end=" ")

print("\n|\_\_\_|\_\_\_|\_\_\_|\_\_\_|")

#####################################################################################################################################

#The Level 3 function

#Executes Level 3

def level3(health,inventory):

#These local variables keep track of whether or not the user

#has done certain things

door="n"

trip="n"

wall="n"

floor="n"

#These are various solutions to the puzzle

puzzle1=["a","b","c","d","a","b","c","d","a","b","c"," ",

"this","serves","to lengthen","the list"]

puzzle2=["a","b","c","d","a","b","c"," ","a","b","c","d",

"this","serves","to lengthen","the list"]

puzzle3=["a","b","c"," ","a","b","c","d","a","b","c","d",

"this","serves","to lengthen","the list"]

puzzle4=[" ","c","b","a","d","c","b","a","d","c","b","a",

"this","serves","to lengthen","the list"]

puzzle5=["d","c","b","a"," ","c","b","a","d","c","b","a",

"this","serves","to lengthen","the list"]

puzzle6=["d","c","b","a","d","c","b","a"," ","c","b","a",

"this","serves","to lengthen","the list"]

#Opening words

print(

"""

You enter the room.

There is a piece of parchment on the east wall, and something metal on the

west wall glints in the beam of your flashlight.

"""

)

#While the user hasn't completed Level 3 or is not dead,

#the user is stuck in Level 3

while door=="n":

#The user can do various things by typing into this input statement

#Depending on whether or not the user has done certain other things,

#only certain things will be availible

user=input("\n>")

user=user.lower()

if (user=="look around" or user=="look"):

print("There is a piece of parchment on the east wall, and something")

print("metal on the west wall glints in the beam of your flashlight.")

if wall=="y":

print("\nIn the hidden room, there is an odd assortment", end=" ")

print("of tiles on the wall.")

#The cheats are so we can get to later levels fast

elif user == 'cheat':

door='y'

elif (user=="inspect floor" or user=="look at floor"):

floor="y"

if trip=="y":

print("\nYou search for the thing", end=" ")

print("that caused you to stumble earlier.")

print("What you find astonishes you: an oddly modern screwdriver.")

else:

print("\nCurious about what might be on the floor,", end=" ")

print("you look by flashlight.")

print("What you find astonishes you:", end=" ")

print("a strangely modern screwdriver.")

elif (user=="pick up screwdriver" or user=="add screwdriver to inventory"

or user=="acquire screwdriver" or user=="take screwdriver"):

if (trip=="n" and floor=="n"):

print("What screwdriver?")

else:

print("\nYou pick up the screwdriver and put it in your pocket.")

inventory+=["screwdriver"]

elif (user=="inspect parchment" or user=="read parchment"

or user=="inspect east wall" or user=="look at east wall"):

if ("screwdriver" not in inventory and trip=="n"):

print("\nOn your way to the wall, you trip over something")

print("small and cyclindrical, and scrape your knee.")

print("It rolls away before you can get a good look at it.")

print("Frustrated, you turn back towards the parchment.")

health-=5

trip="y"

print("\nThe parchment reads:")

print("\neArth beAtS WinD")

print("WisDom is neceSSAry for eScApe.")

print("\nPuzzled, you take the parchment and put it in your pocket.")

inventory+=["parchment"]

elif (user=="look at west wall" or user=="inspect west wall" or

user=="inspect metal"):

if "screwdriver" not in inventory and trip=="n":

print("\nOn your way to the wall, you trip over something")

print("small and cyclindrical, and scrape your knee.")

print("It rolls away before you can get a good look at it.")

print("Frustrated, you turn back towards the wall.")

health-=5

trip="y"

print("\nYou walk to the west wall. There is a")

print("screw drilled into the stone bricks. You wonder")

print("what a modern screw is doing in an ancient pyramid.")

elif (user=="unscrew screw" or user=="take screw" or

user=="use screwdriver"):

if "screwdriver" in inventory:

print("\nYou unscrew the screw, but something", end=" ")

print("happens that you don't expect.")

print("The screw was attached to a string.", end=" ")

print("A weight shifts off of a scale,")

print("and something comes crashing into the wall, knocking it down.")

print("Another room is revealed.")

wall="y"

print("\nUnfortunately for you, a piece of rubble", end=" ")

print("hits you on the head.")

health-=20

print("After rubbing your head, you decide that one such rock")

print("might be effective against enemies- if there were any.")

print("You pick up a chunk of stone and try to put")

print("it in your pocket, but sadly, it does not fit.")

print("You settle with carrying the rock in your other hand.")

inventory+=["rubble"]

else:

print("\nYou try to take the screw out with your fingernail,")

print("but it breaks. Mildly annoyed,", end=" ")

print("you turn away from the wall.")

health-=2

#The tile-slider puzzle

elif (user=="inspect tiles" or user=="solve puzzle"):

#This keeps the user from phasing through walls

if wall=="n":

print("\n...what? There's nothing to see here!")

continue

#These variables keep track of where the blank tile is,

#how many moves the user has made, whether or not they want to

#exit the puzzle, and which tile xe wants to move, respectively

space=11

times=0

wisdom="n"

direct=""

#The current mixed-up start to the puzzle

#The puzzle is printed based upon the order of the letters in

#this list. The list will be edited as the user solves the puzzle

letters=["c","b","c","d","b","d","a","c","a","b","a"," ",

"this","serves","to lengthen","the list"]

display\_puzzle(letters)

print("\nThe tiles appear to be some sort of puzzle.")

#While the user hasn't solved the puzzle, or hasn't stated that

#xe wants to leave the puzzle, the user will be stuck in a loop

while ((letters!=puzzle1 and letters!=puzzle2 and letters!=puzzle3

and letters!=puzzle4 and letters!=puzzle5 and

letters!=puzzle6) and wisdom=="n"):

#The user selects which tile xe wants to move, or if xe wants

#to leave the puzzle. Depending on whether or not xe has

#solved the riddle, xe may be stuck here for a while

direct=input("\n>")

if (direct=="left" or direct=="l" or direct=="a"):

#This extra if statement keeps the tile from

#teleporting to the other side of the puzzle

if (space!=0 and space!=4 and space!=8):

#In the list above, the "space" element is replaced

#by the element it will be replacing

letters[space]=letters[space-1]

#In the list above, the element the space will be

#replacing is replaced by the "space" element

letters[space-1]=" "

#So that the computer knows where the space is,

#the computer adjusts the index number of the space

#based on which direction it was moved

space-=1

display\_puzzle(letters)

#Adds one move to the players total moves

times+=1

else:

display\_puzzle(letters)

continue

elif (direct=="right" or direct=="r" or direct=="d"):

if (space!=3 and space!=7 and space!=11):

letters[space]=letters[space+1]

letters[space+1]=" "

space+=1

display\_puzzle(letters)

times+=1

else:

display\_puzzle(letters)

continue

elif (direct=="up" or direct=="u" or direct=="w" or direct=="top"):

if ((space-4)>-1):

letters[space]=letters[space-4]

letters[space-4]=" "

space-=4

display\_puzzle(letters)

times+=1

else:

display\_puzzle(letters)

continue

elif (direct=="down" or direct=="d" or direct=="s" or direct=="bottom"):

if ((space+4)<12):

letters[space]=letters[space+4]

letters[space+4]=" "

space+=4

display\_puzzle(letters)

times+=1

else:

display\_puzzle(letters)

continue

elif (direct=="wisdom" or direct=="Wisdom"):

wisdom="y"

#If the player makes a typo,

#the computer just displays the puzzle

else:

display\_puzzle(letters)

#If the player broke the loop by solving the puzzle...

if (letters==puzzle1 or letters==puzzle2 or letters==puzzle3

or letters==puzzle4 or letters==puzzle5 or letters==puzzle6):

print("\nIt took you", times, "moves to solve the puzzle.")

print("(in case you were interested)")

print("\nThe door slowly creaks open to reveal", end=" ")

print("another staircase.")

#If the player broke the loop by entering "wisdom"...

elif (direct=="wisdom" or direct=="Wisdom"):

print("\nYou step away from the puzzle.")

else:

print("\nIt seems the gods of the pyramid are", end=" ")

print("conspiring against you.")

print("You decide to try again later.")

elif (user=="climb stairs" or user=="climb staircase" or user=="ascend"):

print("\nYou cautiously ascend the dark stairs.")

door="y"

elif (user=="pick up rubble" or user=="pick up rock"

or user=="acquire rubble" or user=="acquire rock"):

print("You pick up a chunk of stone and try to put")

print("it in your pocket, but sadly, it does not fit.")

print("You settle with carrying the rock in your other hand.")

inventory+=["rubble"]

elif user=="health":

print(health)

elif user=="inventory":

for element in inventory:

print(" -", element)

#In case of invalid choice

else:

number=random.randint(1,3)

if number==1:

print("\n\"", user, "!\" you swear, angry at the pyramid.")

if number==2:

print("\nYou know, \"", user, "\"should be the")

print("first line of your autobiography.")

else:

print("\n\"", user, "!\" you shout, hoping in")

print("vain that it's a password of some sort.")

if health<0:

exit\_room = True

return health,inventory

########################################################################################################################

def level4(health, inventory):

print("\tYou ascend the staircase deliberately, rubbing your head.")

print("\tYou know that there will be a bump there tomorrow.")

#local variables to triger events and locations

door="n"

fanblades=False

wind\_tunnel\_deep=False

wind\_tunnel\_shallow=False

grille\_deep=True

grille\_shallow=True

fan\_placement="wall"

fan\_power=True

while door=="n":

if (wind\_tunnel\_deep==True and wind\_tunnel\_shallow==True):

print("\nThe hidden door between the two grilles opens,")

print("revealing a staircase and a sword.")

print("This sword could replace your rock, which")

print("was never that great of a weapon to start with.")

print("You take the sword and cautiously ascend the stairs.")

inventory.append('sword')

door='y'

user=input("\n>")

user=user.lower()

if user in ("look", "look around"):

print("\nThere are two holes in the south wall,", end=" ")

print("but you can't see them very well")

print("There is also a large tapestry on the north wall.")

print("There is a faint humming sound in the air,")

print("almost like the turning of a small motor.")

print("But you know that can't be.")

print("This pyramid is at least two thousand years old...")

#The cheats are so we can get to later levels fast

elif user == 'cheat':

door='y'

#List all available actions and their consequences

elif user in ("look at wall", "inspect south wall", "inspect wall",

"inspect holes"):

print("\nThere are two holes in the wall, both very deep.")

print("The one on the left is deeper than the one on the right.")

print("Both are covered by metal grilles screwed into the wall.")

print("You see a red flashing light at the end of each.", end=" ")

print("Sensors perhaps?")

print("By now you are used to finding modern technology")

print("in the pyramid and don't find this suspicious.")

elif user in ("listen","inspect motor","inspect air"):

print("\nThe humming sound is coming from the east wall.")

elif user in ("inspect east wall", "inspect humming sound"):

print("\nIn the dark, you almost don't see the fan.")

print("Or at least, you come to the conclusion that the")

print("rapidly rotating rod on a stand is a fan without blades.")

print("But how could it still be running?")

print("You wonder if there are continuity errors in", end=" ")

print("your own expedition.")

elif user in ("inspect north wall", "inspect tapestry"):

print("\nThere is a detailed tapestry hanging on the wall.")

print("You decide that the odd scene it depicts is a grand")

print("royal council made of inhuman figures.")

elif user in ("pull back tapestry", "remove tapestry", "take tapestry"):

print("\nCurious, you pull back the tapestry.")

if fanblades:

print("There is nothing there.")

else:

print("Behind it you find three fan blades.")

elif user in ("take blades", "take fan blades"):

if fanblades:

print("You already have them.")

else:

inventory+=["fan blades"]

fanblades = True

print("\nYou put the fan blades in your pocket,")

print("now very aware of continuity errors.")

elif user in ("inspect fan"):

print("\nYou wonder how the fan has remained spinning all these years.")

print("Unless perhaps the ancients didn't leave so long ago.")

if fan\_power==True:

fan\_power=False

print("\nYou find a blue button at the base of the,", end=" ")

print("fan, and push it, turning the fan off.")

else:

fan\_power=True

print("\nYou find a blue button at the base of the,", end=" ")

print("fan, and push it, turning the fan on.")

elif user in ("push button"):

if fan\_power==True:

fan\_power=False

print("\nYou push the button, turning the fan off.")

if fan\_placement=="deep":

print("\nThe sensor light blinks once, and")

print("you hear another click from behind the wall.")

wind\_tunnel\_deep=False

elif fan\_placement=="shallow":

print("\nThe sensor light blinks once, and")

print("you hear another click from behind the wall.")

wind\_tunnel\_shallow=False

print("But what did the riddle say again?")

else:

fan\_power=True

print("\nYou push the button, turning the fan on.")

if fan\_placement=="deep":

print("\nThe sensor blinks twice, and")

print("you hear a click from behind the wall.")

wind\_tunnel\_deep=True

elif fan\_placement=="shallow":

print("\nThe sensor blinks twice, and")

print("you hear a click from behind the wall.")

wind\_tunnel\_shallow=True

print("But what did the riddle say again?")

elif user in ("attach blades","use blades","attach fan blades",

"use fan blades","insert fan blades","insert blades",

"build fan",'assemble fan'):

if "fan blades" in inventory:

if fan\_power==True:

print("\nYou decide that doing this while the fan is")

print("still running is not a good idea.")

else:

print("\nYou slide the blades into three slots on the rod.")

inventory.remove("fan blades")

else:

print("\nYou don't have any blades to attach.")

elif user in ("acquire fan","take fan","pick up fan"):

if fan\_power==False:

inventory+=["fan"]

print("\nYou put the large fan in your pocket")

print("How exactly you do this is unknown to you.")

fan\_placement="inventory"

else:

print("\nYou decide that putting a spinning fan in your pocket")

print("might shred that lovely tapestry and decide against it.")

elif user in ("use fan","put down fan","place fan","move fan"):

fan\_placement=input("\nWhere? ")

if fan\_placement in ("deep","deep tunnel", "left hole",

"left tunnel","deeper tunnel"):

print("\nYou place the fan in front of the deeper tunnel.")

fan\_placement="deep"

if 'fan' in inventory:

inventory.remove("fan")

if fan\_power==True:

print("\nThe sensor blinks twice, and")

print("you hear a click from behind the wall.")

wind\_tunnel\_deep=True

else:

print("You turn the fan on.")

print("\nThe sensor blinks twice, and")

print("you hear a click from behind the wall.")

wind\_tunnel\_deep=True

fan\_power=True

else:

print("\nThe fan is not in your inventory.")

elif fan\_placement in ("shallow", "shallow tunnel", "right tunnel",

"right hole"):

print("\nYou place the fan in front of the shallow tunnel.")

fan\_placement="shallow"

inventory.remove("fan")

else:

print("\nYou decide that those tunnels have got to do something.")

elif user in ("use screwdriver", "unscrew grille",

"unscrew right grille", "unscrew grilles",

"unscrew screws",'remove grille','remove grilles',

'take grille','take grilles'):

if grille\_deep:

print("You use your screwdriver to remove each grille, setting them asside.")

grille\_deep=False

grille\_shallow=False

elif user in ("throw rock", "throw rubble", "use rubble", "use rock",

"place rubble", "place rock"):

print("\nIf those blinking lights are indeed sensors,")

print("what if one were to break?")

print("\nYou decide that the shallower tunnel is your best bet.")

if grille\_shallow==False:

print("\nYou launch the rock towards the light.")

print("It has the same effect on the sensor as", end=" ")

print("it did on your head.")

print("You hear a click from behind the wall.")

wind\_tunnel\_shallow=True

inventory.remove("rubble")

else:

print("But unfortunately, it is covered by a grille.")

elif user in ("read parchment", "read riddle"):

print("\nYou take the parchment out of your ever-expanding pocket.")

print("It reads:")

print("\neArth beAtS WinD")

print("WisDom is neceSSAry for eScApe.")

elif user in ("ascend stairs", "climb stairs", "climb staircase"):

print("You cautiously walk up the stairs.")

door="y"

break

elif user=="wisdom":

print("\nYou decide to stop breaking the fourth wall, unaware that")

print("by doing this, you have broken the fourth wall again.")

elif user=="health":

print(health)

elif user=="inventory":

for element in inventory:

print(" -", element)

elif user == 'take sword':

inventory.append('sword')

#Deal with invalid commands

else:

number=random.randint(1,3)

if number==1:

print("\n\"", user, "!\" you swear, angry at the pyramid.")

elif number==2:

print("\nYou know, \"", user, "\" should be the")

print("first line of your autobiography.")

else:

print("\n\"", user, "!\" you shout, hoping in")

print("vain that it's a password of some sort.")

if health < 0:

door='y'

return health, inventory

#########################################################################

def level5(health, inventory):

#Some local variables

win = False

vault = True

###########Computer Function##############

def computer(vault):

command = None

password = 'aliens'

while command!=password:

command=input("\t\tPassword (who built the pyramids?): ")

cont = True

print("""

0 - quit

1 - open vault

""")

while cont:

command = input("\t\tc:\\>")

if command == '0':

cont = False

elif command == '1':

if vault:

print("\t\tVault security deactivated.")

vault = False

else:

print("\t\tVault security already deactivated.")

return vault

#More local variables

gas = False

alien = True

print("You step into a modern laboratory.")

print("Two sliding doors seal behind you silently, apparently dissapearing.")

exit\_room = False

mask = False

while not exit\_room:

action = input("\n> ")

action = action.lower()

#Calculates remaining time IF gas attack has started

if gas:

t = time.time()

te = t - bt

tr = 120 - te

#Decides whether user is alive or not

if tr < 0:

health = -1

print("Suddenly you crumple into a heap on the floor, unable to take another breath.")

else:

print("\nYou have",tr,"seconds left")

if health<0:

exit\_room=True

#Available actions

if action == 'inventory':

for element in inventory:

print(" -", element)

elif action == '':

None

elif action in ('inspect alien','search alien'):

if alien:

print("That alien is pretty big.")

print("You had better deal with him.")

else:

if 'gas mask' not in inventory:

print("You search the dead alien.")

print("You take a gas mask that was on him.")

inventory.append('gas mask')

else:

print("The alien won't be a problem anymore.")

elif action == 'health':

print(health)

elif action == 'look':

if alien:

print("In front of you, an alien is typing away at a computer terminal. He is unaware of your presence.")

else:

print("The alien lies dead on the floor.")

if 'gas mask' not in inventory:

print("You notice he is wearing somthing...")

if vault:

print("A large vault lies beyond the computer terminal.")

else:

print("The vault door is open")

elif action in ('attack alien', 'kill alien'):

if alien:

print("You sneak up behind the alien and draw your sword.")

print("In one swift movement you slice the alien in two.")

alien = False

else:

print("The alien is already dead.")

elif action in ('inspect computer','inspect terminal','use computer','use terminal'):

if alien:

print("That's probably not a good idea,")

print("seeing as how there is a large alien right there.")

#Computer function. Allows player to use computer.

else:

vault=computer(vault)

elif action in ('inspect vault','enter vault','open vault'):

if vault:

print("The vault is locked.")

else:

print("You step into the vault. There is enough gold to last you a lifetime!.")

#Trigger gas attack if it hasn't already started and user isn't wearing a gas mask

if not gas and not mask:

print("Unfortunately, you seem to have steped on a pressure")

print("plate. A poisonous gas starts pouring through vents in the celing.")

gas = True

bt = time.time()

print("you estimate that you have two minutes before your lungs fail.")

elif action in ('wear gas mask','use gas mask','wear mask','use mask'):

if 'flashlight' in inventory:

print("The mask requires some sort of power source.")

print("You must have some batteries on you somewhere!")

else:

print("You quickly put on the mask, and feel the rush of clean air into your lungs.")

mask = True

gas = False

elif action == 'open flashlight':

if 'flashlight' in inventory:

print("You remove the back of your flashlight, revealing two AA batteries.")

print("You quickly put them in your gas mask. It should work now!")

inventory.remove('flashlight')

else:

print("You don't have a flashlight.")

elif action == 'take gold' and not vault:

#Player must survive first

if gas:

print("You have a bigger problem to worry about right now!")

#Trigger victory

else:

print("You fill your pockets with as much gold as possible.")

print("You have finally obtained riches beyond your wildest dreams!")

win = True

exit\_room = True

#Deal with invalid answer choices

else:

if alien:

print("You scream", action, ",hoping that it is some kind of password. Somehow, the alien fails to notice your blood curdling scream. Scream is also a painting by edouard manet. Perhaps you too will be a famous painter after you discover the treasure.")

else:

print("You scream", action, ",hoping that it is some kind of password.")

return health,inventory,win

# Main block

#While the player isn't dead and hasn't won yet

while health>0 and not win:

if level==0:

introduction()

elif level==1:

health,inventory = level1(health,inventory)

elif level==2:

health,inventory = level2(health,inventory)

elif level==3:

health,inventory = level3(health,inventory)

elif level==4:

health,inventory = level4(health,inventory)

elif level==5:

health,inventory,win = level5(health,inventory)

#Progress to the next level

level+=1

#Alternate ending statements

print("\n\n")

if win:

print("Congratulations, treasure hunter! You have completed your quest.")

else:

print("Your injuries are too severe. You die alone, forever entombed in this mysterious pyramid.")

input("Press the enter key to exit.")