International Journal of Applied Science and Mathematics
Volume 9, Issue 1, ISSN (Online): 2394-2894

On the Topological Sets of Larger Cardinalities

David Delali Zigli[3*], Vivian Maanu[1], Samuel Amoako[2] and Prince Nimako Gyasi[3]
[3] Department of Mathematical Sciences, University of Mines and Technology, Tarkwa, Ghana.
[1, 2] Department of Mathematics, Akrokerri College of Education, Akrokerri, Ghana.
*Corresponding author email id: davidzigli@gmail.com

Date of publication (dd/mm/yyyy): 05/03/2022

***Abstract* – Topology is one of the active areas of all mathematics. Conventionally, it is considered one of the three areas of pure mathematics, together with algebra and analysis. Formally, if X is a set and τ a family of subsets of X, then τ is a topology on X if both the empty set and X are elements of, τ, any union of elements of τ is an element of, τ and any intersection of finitely many elements of τ is an element of τ. However, checking the topology of a set with cardinality manually is difficult and time-consuming. This study, therefore, introduced a technologically inclined approach (the Python programming language) with a well-developed user-interface for checking the topology of a set or sets with cardinality. The programming language (Python code) was developed using principles of set theory and the definition of the topology of a set. The friendly user-interface developed with the embedded algorithm is able to determine the topology of a set with a larger cardinality and also extract the open and closed sets of the topology, hence making it faster and cheaper.**

***Keywords* – Topological Set, Open Set, Cardinality, Programming, Elements.**

I. Introduction

Topology is a branch of mathematics that is used in real-world situations [18]. The study of shapes, including their properties, deformations applied to them, mappings between them, and configurations made up of them, is known as topology [6]. This is why topology is so important for anyone aspiring to be a mathematician. Furthermore, because topology is an abstract concept, discussing the unknowns makes mathematics appear to be a difficult subject for everyone [16]. Other studies were successful in explaining related concepts like metric space, hamming distance, and Levenshtein distance using real-world examples. The applicability of metric space was also demonstrated using DNA hybridity, [20]. Amoeba Proteus, a single-celled organism, was also successfully used to demonstrate the use of Homeomorphism [19]. However, in all these studies, the topological concepts used were checked using the traditional time-consuming approach. So, determining the topology, openness and closeness of a set continue to be a source of anxiety for students. Due to the difficulties in manually detecting the topology of sets, most students' and mathematicians' first loves are or will be algebra, analysis, geometry, mathematical biology, operation research, or statistics. As a result, topology must be made simple and interesting in order for them to consume it. Manually checking the topology of a set with cardinality $n \in Z^+$ is difficult and time-consuming. We may be unable to determine whether or not a set A of 250 elements is a topology on X of 500 elements. As a result, this project took a more advanced approach to problem solving by writing codes (in a computer programming language) to check the topology of sets, as well as their closeness and openness. However, when compared to the manual method of checking for topology of sets, this will simplify and make the detection or checking of a topology of sets simple and interesting, facilitating teaching and learning of topology of sets. This work aims to use Python code based on a topology's definition or axioms to determine the topology of a given set or sets with larger cardinality $n \in Z^+$.

II. Preliminaries

2.1. *Set Theory*

Georg Cantor (1845), a German mathematician, invented "the theory of sets" or "Set Theory." He came across sets while working on "Problems on Trigonometric Series," which have since become one of mathematics' most fundamental concepts. If you do not understand sets, it will be difficult to explain other concepts such as relations, functions, sequences, probability, geometry, and so on.

Definition 2.1.1

A set is a clearly defined group of objects or people. Sets can be related to a variety of real-world examples, such as the number of rivers in Ghana, the number of colours in a rainbow, and so on.

Definition 2.1.2

If $n \in Z^+$ for some class *Y*, then *X* is a set. If $X \in Y$ for any class *Y*, then *X* is a proper class. You will notice immediately that a set is exactly an element. This is the awkward aspect of our terminology: An axiomatic set theory requires us to accept the words "element" and "set" as synonyms, to be distinguished from proper classes, which are not sets, and cannot be elements of anything. Proper classes are a little like embarrassing relatives that we are forced to acknowledge, but we'd rather keep at arm's length. Those are the classes that may lead to paradox, and in order to do mathematics safely we want to be sure that we are dealing only with sets. So, the most important axioms of set theory are designed to provide the assurance that when we carry out operations on sets, for example when we form a generalized union of sets, we are not inadvertently creating a proper class [2].

To be precise, we want a guarantee that when performing operations on sets the results of the operations are sets and not proper classes. Since, intuitively, a set is any class that is not "too large", we would certainly expect every subclass of a set to be a set. So, if *A* is not too large and $B \subseteq A$, then *B* should not be too large.

Proposition 2.1:

Every subclass of a set is also a set.

Proof:

If $A \cap B \subseteq A$. Then *A* is a set, then $A \cap B$ is a set. That is, the intersection of any two sets is a set. With all this talk about sets, one would assume that we could exhibit one in other words, we could say, here, this is a set! Mighty oaks from little acorns grow. We shall see later that from the humble empty set, {} many more sets can be shown to exist. Consider the set {}: It has one element, namely the empty set. The set {{a}, {}} has two elements. It is reasonable to assume that if *a* and *b* are sets, then the doubleton {a, b}, with only two elements, is not too large to be a set. Also, if *a* and *b* are sets, then {a, b} is a set. It therefore follows immediately that if a is a set, then the singleton {a} is a set. The intersection is especially important, because they guarantee that if you combine sets into larger sets, for example by forming the generalized union of a family of sets, or a Cartesian product of many sets these larger collections are still sets [8].

Definition 2.1.3

Let *A* be a set; by the power set of *A* we mean the class of all the subsets of *A*. In symbols, the power set of *A* is the class, is the class of all the sets *B* which satisfy $B \subseteq A$.

Example 2.1.4

If A = {a, b}, then note that if and only if $B \subseteq A$., It is easy to see that is a larger class than A, for includes all the singletons {x} as x ranges over A. Thus, we may legitimately ask the following question: If A is a set, is it necessarily true that is a set? Or is it possible that may be too large to be a set? An analogous question may be raised in regard to the union of sets: if is a set of sets, is a set, or might it be too large to be a set? These questions may be answered intuitively as follows: None of the "giant" collections which cause contradictions in intuitive set theory can be obtained either as a power set of a set or as a union of a set of sets. Thus, we are justified in adopting the following as axioms. If A is a set, then power set of A is a set. If A and B are sets, then by set theory {A, B} is a set.

2.2. *Topological Space*

A topological space is a generalization of the notion of an object in three-dimensional space. It consists of an abstract set of points along with a specified collection of subsets, called open sets that satisfy three axioms, namely;

- The set itself and the empty set are open sets;
- The intersection of a finite number of open sets is open; and
- The union of any collection of open sets is an open set.

Example 2.2.1

Let X ={a, b, c, d, e, f} and τ_1={X, $\emptyset$,{a},{c, d},{a, c, d},{b, c, d, e, f}}. Then τ_1 is a topology on X as it satisfies conditions (i), (ii) and (iii) of Definitions.

Definition 2.2.1

Let (X, τ) be any topological space, then the members of τ are said to be open sets, [18].

Example 2.2.2

That, thus, $A \cup B$is a set. This shows that the union of two sets is a set. Several other axioms for sets have been proposed, but are not essential in everyday mathematical practice. One axiom, that we shall encounter again, is called the "Axiom of Foundation". It states the following: If A is any set, there is an element $a \in A$such that$a \cap A = \{\}$. This axiom has an equivalent form which has applications in real life situations. Any descending sequence of sets, say $D \in C \in B \in A$ is finite.

In other words, you cannot have an infinite descending sequence of sets, each an element of the previous one. A very intriguing axiom, which has surprisingly far-reaching consequences is proper class, is in one-to-one correspondence with the universal class that is, with the class of all sets [10]. Though we shall not adopt this axiom here, it helps us to form an intuitive image of what proper classes are like.

2.3. *Open and Closed Sets*

Definition 2.3.1

Let (X, τ) be any topological space. Then the members of τ are said to be open sets.

Proposition 2.3.1

If (X, τ) is any topological space, then,

(i) X and Ø are open sets,

(ii) The union of any (finite or infinite) number of open sets is an open set, and

(iii) The intersection of any finite number of open sets is an open set.

Definition 2.3.2

Let (X, τ) be a topological space. A subset S of X is said to be a closed set in (X, τ) if its complement in X, namely X\S, is open in (X, τ).

Proposition 2.3.2

If (X, τ) is any topological space, then

(i) Ø and X are closed sets,

(ii) The intersection of any (finite or infinite) number of closed sets is a closed set and

(iii) The union of any finite number of closed sets is a closed set.

Proof:

It follows immediately from *Proposition 2.3.*1 and *Definition 2.3.2*, as the complement of X is $\emptyset$ and the complement of $\emptyset$ is X. To prove that (iii) is true, let $S_1, S_2,...,S_n$ be closed sets. We are required to prove that $S_1 \cup S_2 \cup \cdots \cup S_n$ is a closed set It suffices to show, by *Definition 2.3.1*, that $X \setminus (S_1 \cup S_2 \cup \cdots \cup S_n)$ is an open set. As $S_1, S_2,...,S_n$ are closed sets, their complements $X \setminus S_1, X \setminus S_2,...,X \setminus S_n$ are open sets. But $X \setminus (S_1 \cup S_2 \cup \cdots \cup S_n) = (X \setminus S_1) \cap (X \setminus S_2) \cap \cdots \cap (X \setminus S_n)$.

As the right-hand side of (1) is a finite intersection of open sets, it is an open set. So, the left-hand side of (1) is an open set. Hence $S_1 \cup S_2 \cup \cdots \cup S_n$ is a closed set, as required. So (iii) is true. The proof of (ii) is similar to that of (iii).

Theorem 2.3.1

Let A be a subset of a topological space X, and let A' be the set of limit points of A. Then C1(A) = $A \cup A'$ [17].

Proof:

We show that $A \cup A' \in$ C1(A) and Cl(A) $\in A \cup A'$. First, we show that $A \cup A' \in$ Cl(A). Certainly, $A \in$ Cl(A), so all we need to show is that $A' \in$ Cl(A). Suppose $X \in A'$. Then every neighbourhood of X intersects A. It hence follows that $A' \in$ Cl(A). Thus $A \cup A' \in$ Cl(A). In the former case, it follows that $X \in A \cup A'$. Since $X \in$ Cl(A), the theorem above implies that every open set containing X intersects A. Since X of A, such an intersection must contain a point other than X. Thus, X is a limit point of A, and it follows that $X \in A \cup A'$. In either case ($X \in A$ or $X \in$ C1(A)) we have $X \in A \cup A'$, implying that Cl(A) $\in A \cup A'$ [3].

Corollary 2.3.1

A subset A of a topological space is closed if and only if it contains all of its limit points [14].

Proof:

By theory A is closed if and only if A = Cl(A) if and only if $A = A \cup A'$ where A' is the set of limit points of A. Finally, $A = A \cup A'$ holds if and only if $A' \in A$. Thus, A is closed if and only $A' \in A$.

2.4. *Interior, Closure, and Boundary*

An arbitrary subset A of a topological space might be neither open nor closed. However, it is often useful to associate a related open set or a related closed set to A. In particular, we can sandwich each set A between the largest open set contained in A and the smallest closed set containing A. These sets are known as the interior of A and the closure of A, respectively [1]. A fundamental defining property of the real number system is the least upper bound property: Every subset of the real number system R, that is bounded from above has a least upper bound. Equivalent to this property is the greatest lower bound property: Every subset of real number system R, that is bounded from below has a greatest lower bound. We denote the least upper bound and greatest lower bound of a set A by lub(A) and glb(A), respectively.

Definition 2.4.1

Let A be a subset of a topological space X. The interior of A, denoted A or lnt(A), is the union of all open sets contained in A. The closure of A, denoted A or Cl(A), is the intersection of all closed sets containing A, [1]. Clearly, the interior of A is open and a subset of A, and the closure of A is closed and contains A. Thus, we have the fore mentioned set sandwich, with A caught between an open set and a closed set: A $\subset$ A $\subset$ A. The following properties follow readily from the definition of interior and closure.

Theorem 2.4.1

Let X be a topological space and A and B be subsets of X:

i. If U is an open set in X and U $\subset$ A, then U $\subset$ Int(A).

ii. If C is a closed set in X and A $\subset$ C, then Cl(A) $\subset$ C.

iii. If A $\subset$ B then Int(A) $\subset$ Int(B).

iv. If A $\subset$ B then Cl(A) $\subset$ Cl(B).

v. A is open if and only if A = Int(A).

vi. A is closed if and only if A = Cl(A), [12].

Proof:

From the above theorems, suppose that U is an open set in X and U $\subset$ A. Since Int(A) is the union of all of the open sets that are contained in A, it follows that U is one of the sets making up this union and therefore is a subset of the union. That is, U $\subset$ Int(A). Since A $\subset$ B, Int(A) is an open set contained in B. This implies that every open set contained in B is contained in Int(B). Therefore Int(A) $\subset$ Int(B). Again, if A = Int(A), then A is an open set, since by definition Int(A) is an open set. Now assume that A is open. We show that A = Int(A). First, Int(A) $\subset$ A by definition of Int(A). Furthermore, since A is an open set contained in A, it follows that A $\subset$ Int(A). Thus A = Int(A) as we have shown [11].

Theorem 2.4.2

Every open set contained in A is contained in the interior of A. In this way Int(A) is the largest open set con-

-tained in A. Similarly, every closed set containing A also contains the closure of A, and thus Cl(A) is the smallest closed set containing A.

III. Results and Discussion

3.1. *Basis for Topology*

Let B be a basis on a set *X*. The topology τ generated by B is obtained by defining the open sets to be the empty set and every set that is equal to a union of basis elements [9]. We need to check that the resulting collection τ is actually a topology; we do that after looking at a couple of examples.

Definition 3.1.1

Suppose *X* is a non-empty set, a collection τ of subset of X is said to be a topology on X, if:

i. X and the empty set, belongs to τ.

ii. The union of any (finite or infinite) number of sets in τ belongs to τ.

iii. The intersection of any two sets in τ belongs to τ. Where τ represent topology on X and X represent the non-empty set. The non-empty set X together with a topology τ on X is called a topological space [7].

Proposition 3.1.2

The topology τ generated by a basis *B* is a topology [7].

Proof:

The empty set {}, is in τ by definition. Since every point in X is contained in some basis element, it follows that X is the union of all of the basis elements and therefore is in τ. Next, we show that a finite intersection of sets in τ is in τ. Also assume that each is a union of basis elements. We show that is a union of basis elements. Finally, we show that an arbitrary union of sets in τ is in τ. Therefore, a collection of sets τ is a topology, and we are justified in calling it the topology generated by the basis, B [12].

Proposition 3.1.3

If (X, τ) is a topological space such that for every $x \in X$, the singleton set $\{x\}$ is in τ, then τ is the discrete topology [13].

Proof:

Every set is a union of its singleton subset. Let *S*, be a subset of *X*. Then S = $U_{x \in s\{x\}}$, since {x} is in τ, *proposition 3.1.3* above imply that $S \in$ τ. As *S* is an arbitrary subset of *x*, we have that τ the discrete [13].

3.2. *Cardinality*

The cardinality of a set is a measure of the set's "number of elements" in mathematics. For example, the set A = {2, 4, 6,} contains three elements, so A has a cardinality of three. Beginning in the late nineteenth century, this concept was generalized to infinite sets, allowing one to distinguish between different types of infinity and perform arithmetic on them. There are two approaches to cardinality: one that compares sets directly using bijections and injections, and another that uses cardinal numbers. When there is no possibility of confusion with

International Journal of Applied Science and Mathematics
Volume 9, Issue 1, ISSN (Online): 2394-2894

other notions of size, the cardinality of a set is also known as its size.

3.3. *Test of Developed Code*

In this section, the written python code is tested with different instances. The code will be referred to as “Topochecker” meaning Topology checker. See Appendix I for part of the developed code.

3.4. *Case 1*

Suppose x is a set in a topological space (X, τ), when *x* is a non-empty set with the element: X = {a, b, c, d, e, f}, τ = {X, {}, {a}, {c, d}, {a, c, d}, {b, c, d, e, f}} where τ represent topology on x, and also satisfies all the axioms and Conditions.

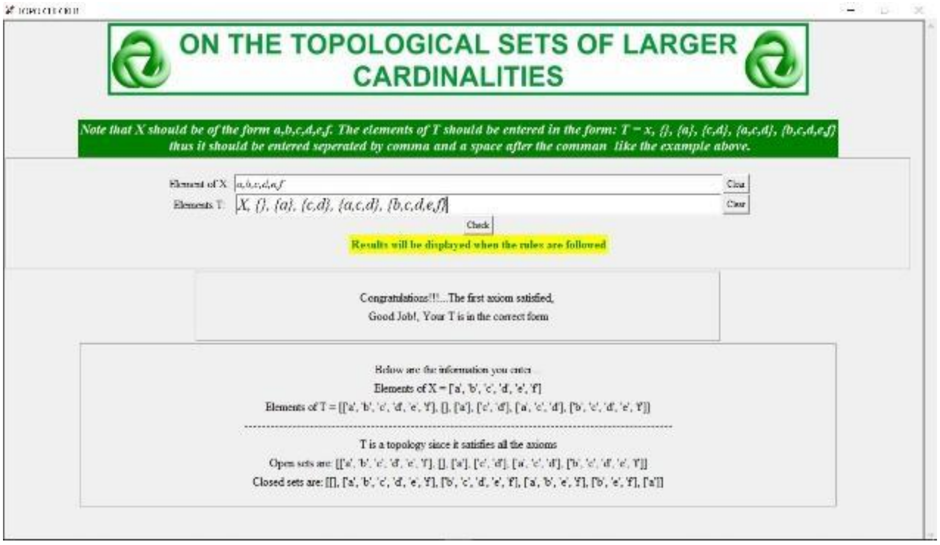

Fig. 3.1. Applying Topochecker to Case Study 1.

From the outcome, of the topochecker: The first axiom was satisfied since, *x* and {} belongs to τ, so the topochecker interface displayed the set $X = \{a, b, c, d, e, f\}$ thus the elements we input with its associated $\tau = \{x, \{\}, \{a\}, \{c, d\}, \{a, c, d\}, \{b, c, d, e, f\}\}$. The second and the third axiom was also satisfied, since the intersection of:

$$X \cap \{\} = \{\}.$$

$$\{c, d\} \cap \{a, c, d\} = \{c, d\},$$

$$\{a, c, d\} \cap \{b, c, d, e, f\} = \{c, d\}.$$

These results were all found in the τ. Thus to say that the intersection of any set in τ belongs to τ. Also, the union of: $\{a\} \cup \{c, d\} = \{a, c, d\}$ $\{c, d\} \cup \{a, c, d\} = \{a, c, d\}$. $\{a, c, d\} \cup \{b, c, d, e, f\} = \{b, c, d\}$. Again, all the results were found in the τ. Thus, to say that the union of any set in τ belongs to the τ.

3.5. *Case 2*

Let *x* be a set in a topological space (X, τ), when *x* is a non-empty set with the element: $X = \{a, b, c, d, e\}$ $\tau = \{X, \{\}, \{a\}, \{c, d\}, \{a, c, e\}, \{b, c, d, e\}\}$ where τ represent topology on *x*, and also satisfies all the axioms and conditions. From the outcome, of the topochecker: The various reasons behind how the given set was not a topology was well outlined, as detected by the topochecker.

International Journal of Applied Science and Mathematics
Volume 9, Issue 1, ISSN (Online): 2394-2894

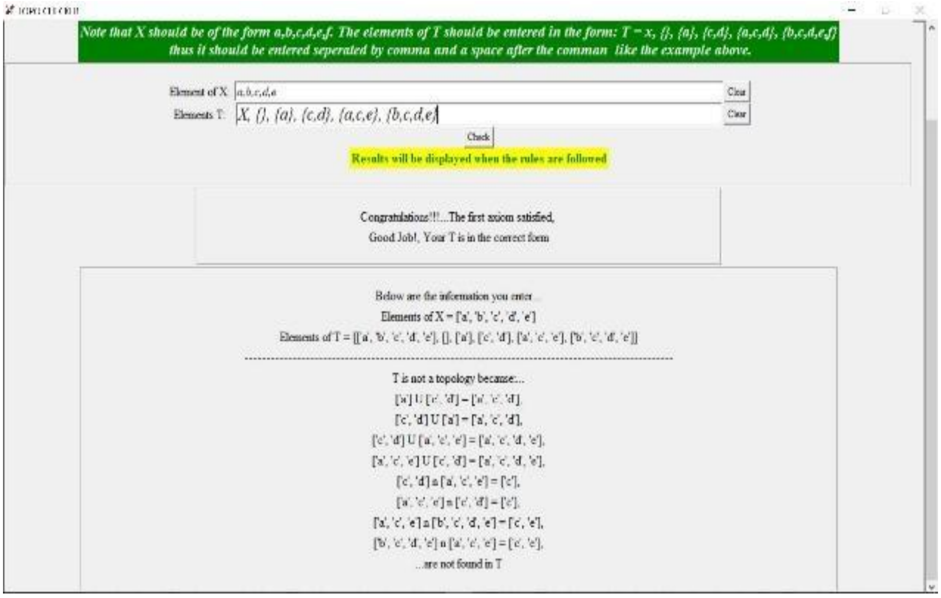

Fig. 3.2. Applying Topochecker to Case Study 2.

3.6. *Case 3 (Error Detecting)*

Let x be a set in a topological space (X, τ), when x is a non-empty set with the element: $X = \{a, b, c, d, e, f\}$ $\tau = \{X, \{\}, \{-\}, \{+\}, \{=\}\}$ Where τ represent topology on x, and also satisfies all the axioms and conditions. From the outcome, of the topochecker, the empty set and X which is always in the τ was satisfied since it is a trivial axiom to the topology. Moreover, it reflects how the topochecker works when you enter the sets in a different order or if the entries are not in line with what is prescribe by the programmer.

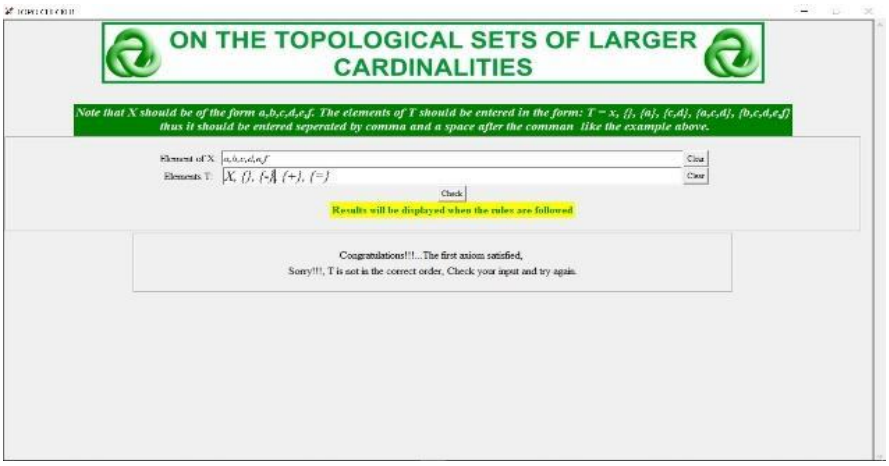

Fig. 3.3. Applying Topochecker to Case Study 3.

IV. Conclusions and Recommendation

4.1. *Conclusions*

The topochecker results for Case 1 were correct because they correctly checked the fundamental axioms of topology and listed the open and closed sets. In Case 2, the code determined that the set does not have a topology with reasons. In Case 3, it reflects how the topochecker works correctly when the sets are entered in a different order or when the entries are not in the order specified by the programmer.

It is therefore advised that, in order to obtain the desired result, a user should take care to follow the simple right order of set entries. It is strongly recommended that all entries be done correctly, patiently, and accurately in order to produce accurate results. To maintain good interaction, it is recommended that a given set have a topology before the topochecker can classify it as an open or closed set.

It is strongly advised that students work on projects to develop smarter or more technologically oriented techniques for determining other topological concepts such as limit points, closures, interior points, and so on.

APPENDIX I

```
from tkinter import *
from PIL import ImageTk, Image
from tkinter import ttk

root = Tk()
root.geometry("1100x480")

#Creating the Title of the Program
root.title('TOPO CHECKER')
root.iconbitmap('icon2.ico')

# Create A Main Frame
main_frame = Frame(root)
main_frame.pack(fill=BOTH, expand=1)

# Create A Canvas
my_canvas = Canvas(main_frame)
my_canvas.pack(side=LEFT, fill=BOTH, expand=1)

# Add A Scrollbar To The Canvas
my_scrollbar = ttk.Scrollbar(main_frame, orient=VERTICAL, command=my_canvas.yview)
my_scrollbar.pack(side=RIGHT, fill=Y)

# Configure The Canvas
my_canvas.configure(yscrollcommand=my_scrollbar.set)
my_canvas.bind('<Configure>', lambda e: my_canvas.configure(scrollregion = my_canvas.bbox("all")))

# Create ANOTHER Frame INSIDE the Canvas
second_frame = Frame(my_canvas)

# Add that New frame To a Window In The Canvas
my_canvas.create_window((0,0), window=second_frame, anchor=NW)
```

```
#Inserting the UMaT header
my_img = ImageTk.PhotoImage(Image.open('umat_header.jpg'))
my_header = Label(second_frame,image = my_img, anchor = CENTER).pack()
#Creating and Displaying the Welcome to TOPO CHECKER
L1 = Label(second_frame, text = ' ',fg = 'blue', font = 'Times 16 bold',anchor = N)
L1.pack()#grid(row = 0, column = 0)
#Creating and displaying the caption for the program

L3 = Label(second_frame, text = 'Note that X should be of the form a,b,c,d,e,f. The elements of T should be entered in the form: '
            'T = x, {}, {a}, {c,d}, {a,c,d}, {b,c,d,e,f}'
           '\n thus it should be entered seperated by comma and a space after the comman '
           ' like the example above.'
           , bg = 'green',fg ='white',font = 'Times 14 italic bold',anchor = N
            ).pack()

#Creating the second frame
frame2 = LabelFrame(second_frame, padx = 220,pady = 20)
frame2.pack(padx = 1, pady = 1)
#A frame accepting the X input for processing
L4 = Label(frame2, text = 'Element of X: ', font = 'Times 12').grid(row = 0, column = 0)
entry_x = Entry(frame2, font ='Times 12 italic',width = 83, borderwidth = 2)
entry_x.grid(row = 0, column = 1)
#A frame accepting the T input for processing
L5 = Label(frame2, text = 'Elements T: ',font = 'Times 12').grid(row = 1, column = 0)
entry_T = Entry(frame2,font ='Times 16 italic', width = 60, borderwidth = 2)
entry_T.grid(row = 1, column = 1)
Ls1 = Label(frame2, text='Results will be displayed when the rules are followed'
       ,font = 'Times 12 bold', bg = 'yellow', fg = 'green').grid(row=4, column=1)
#Functions to clear input words when the clear button is clicked
def clear1():
    entry_x.delete(0,END)

def clear2():
    entry_T.delete(0,END)

#Creating the clear buttons
b1 = Button(frame2, text = 'Clear', font ='Times 9',command = clear1).grid(row=0,column = 2)
b2 = Button(frame2, text = 'Clear', font = 'Times 9', command = clear2).grid(row=1,column = 2)
```

International Journal of Applied Science and Mathematics
Volume 9, Issue 1, ISSN (Online): 2394-2894

```
def check():
    frame_s = LabelFrame(second_frame, padx=220, pady=20)
    frame_s.pack(padx=1, pady=1)
    x = []
    for i in entry_x.get():
        i = i.strip()
        if i != ',':
            if i!='':
                x.append(i)

    x = set(x)
    print(type(x))
    print(sorted(list(x)))

    S = entry_T.get()
    a = 0
    T = []
    count_s = S.count(', ')
    curly = 0
    b = 0
    if '{}' in S:
        curly = 1
    n = [',  ', ',   ',',    ',' ',' ',' ',' ',  ', '[',']', '.', '. ', ' .', ':','|', '+', '-', '=', '&', '%','$', '@', '!', '^', '*', '?', '<','>','\/', '/']
    for i in n:
        if i in S:
            b = 20

    if 'x' in S or 'X' in S:
        if '()' in S or '{}' in S:
            NL = Label(frame_s, text='Congratulations!!!...The first axiom satisfied, '
                       , font='Times 12', anchor=N).grid(row=0, column=0)
            print('First axiom satisfied')
            T.append(x)
            T.append(set())
            a = 1
    else:
        NL = Label(frame_s, text='Sorry!!! your T is not a topology since '
                            'X and {} are not found in T, '
                   , font='Times 12', anchor=N).grid(row=1, column=0)
```

```
    print('Sorry!!! your T is not a topology since X and {} are not found in T , ')
if a == 1:
    s = 0
    s= S.split(', ')
    print ('s ', s)
    for i in s:
        if i != 'X' and i != 'x':
                if i != '()' and i != '{}':
                        r = i.strip()
                        r = i.strip('{}')
                        print ('r ',r)
                        u = []
                        e = r.split(',')
                        for i in e:
                            u.append(i.strip())
                        print (e)

                        T.append(set(u))
    U = []
    for i in T:
        U.append(sorted(i))

    if curly == 1:
        if S.count('{') +1 != len(T) or b ==20:
            a = 20
            NL = Label(frame_s, text='Sorry!!!, T is not in the correct order, '
                                    'Check your input and try again.'
                        ,font='Times 12', anchor=N).grid(row=2, column=0)
            print('Sorry!!!, T is not in the correct order, Check your input and try again.')
    else:
        if S.count('{') + 2 != len(T) or b ==20:
            a = 20
            NL = Label(frame_s, text='Sorry!!!, T is not in the correct order, '
                                    'Check your input and try again.'
                        , font='Times 12', anchor=N).grid(row=2, column=0)
            print('Sorry!!!, T is not in the correct order, Check your input and try again.')
    if a != 20:
        NL = Label(frame_s, text='Good Job!, Your T is in the correct form'
                    , font='Times 12', anchor=N).grid(row=3, column=0)
        print('Your T is in the correct form')
```

```
frame3 = LabelFrame(second_frame, padx=220, pady=20)
frame3.pack(padx=1, pady=1)
L6 = Label(frame3,text = 'Below are the information you enter...',font = 'Times 12')
L6.grid(row = 0, column = 0)
L7 = Label(frame3,text = 'Elements of X = {}'.format(sorted(list(x))),font = 'Times 12')
L7.grid(row = 1, column = 0)
L8 = Label(frame3, text='Elements of T = {}'.format(U), font = 'Times 12')
L8.grid(row=2, column=0)

#L6 = Label(frame2, text ='Loading...').grid(row = 3, column = 1)

i = 0

S_u = []
S_i = []
info_u = []
info_n = []
while i < len(T):
   j = 1
   while j < len(T):
      a = T[i] | T[j]
      w1 = '{} U {}'.format(sorted(T[i]), sorted(T[j]))
      info_u.append(w1)
      S_u.append(sorted(a))
      b = T[i] & T[j]
      w2 = '{} n {}'.format(sorted(T[i]), sorted(T[j]))
      info_n.append(w2)
      S_i.append(sorted(b))
      j += 1
   i += 1

n = None
total = S_u + S_i
total_n = info_u + info_n

i = 0
T_new = []
for i in T:
   T_new.append(sorted(list(i)))
```

```
chaf = []
chaf_info = []
q = 0
c = 5
Ls1 = Label(frame3, text='-----------------------------------------------'
            '-----------------------------------------------',font = 'Times 12').grid(row=3, column=0)
for i in total:

    if i not in T_new:
        chaf.append(i)
        chaf_info.append(total_n[q])
        n = 2
    q += 1
if n == 2:

    L9 = Label(frame3, text = 'T is not a topology because:...',font = 'Times 12')
    L9.grid(row = 4, column = 0)
    print('T is not a topology because:...')
    i = 0

    while i < len(chaf):
        L10 = Label(frame3, text = '{} = {},'.format(chaf_info[i], chaf[i]),font= 'Times 12')
        L10.grid(row = c, column = 0)
        print('{} = {},'.format(chaf_info[i], chaf[i]))
        i += 1

        c +=1
    L11 = Label(frame3, text = '...are not found in T', font = 'Times 12' )
    L11.grid(row = c, column =0)
    print('...are not found in T')
else:
    L12 = Label(frame3, text = 'T is a topology since it satisfies all the axioms', font = 'Times 12')
    L12.grid(row = c+1, column = 0)
    print('T is a topology since it satisfies all the axioms')
    print()

    x_new = list(x)
    x_new.append([])
    m = None
    open_set = []
```

```
        for i in T_new:
            m = None
            for a in i:
                if a not in x_new:
                    m = 2
            if m == 2:
                print(i, 'is not opened')
            else:
                open_set.append(i)

        print()
        L13 = Label(frame3, text = 'Open sets are: {}'.format(open_set),font = 'Times 12')
        L13.grid(row = c+2, column = 0)
        print('Open sets are: ', open_set)
        print()
        closed_set = []
        for i in open_set:
            closed = x.difference(set(i))
            closed_set.append(sorted(closed))
        L14 = Label(frame3, text = 'Closed sets are: {}'.format(closed_set),font = 'Times 12')
        L14.grid(row = c+3,column = 0)
        print('Closed sets are: ', closed_set)

b3 = Button(frame2, text='Check', font = 'Times 9',command = check).grid(row=2, column=1)

root = mainloop()
```

REFERENCES

[1] Armstrong, M. A. (2013), *"Basic topology"*, Springer Science & Business Media.

[2] Arnold, B. H. (1949), "A topological proof of the fundamental theorem of algebra", *The American Mathematical Monthly*, Vol. 56, No. 7, pp. 465-466.

[3] Barmak, J. A. (2011), *"Algebraic topology of finite topological spaces and applications"*, Vol. 2032. Springer.

[4] Brouwer, L. (1910), "On the structure of perfect sets of points", In *KNAW, Proceedings*, Vol. 12, pp. 785-794.

[5] Cannon, J. W. and Conner, G. R. (2006), "On the fundamental groups of one-dimensional spaces", *Topology and its Applications*, Vol. 153, No. 14, pp. 2648-2672.

[6] Carlson, S.C., (2017), "Topology", *Encyclopedia Britannica*, available at: http://www.britannica.com/science/topology. Accessed: November, 9, 2021.

[7] Colin C. Adams. The knot book: A elementary introduction to the mathematical theory of knots. Freeman and Co., New York, 1994.

[8] Dodson, C. T., Parker, P. E., and Parker, P. (1997), *"A user's guide to algebraic topology"*, Vol. 387. Springer Science & Business Media.

[9] Dorfer, G., Thuswaldner, J. M., and Winkler, R. (2013), "Fundamental groups of one-dimensional spaces", *Fund. Math*, Vol. 223, pp. 137-169.

[10] Eda, K. and Kawamura, K. (1998), "The fundamental groups of one-dimensional spaces", *Topology and its Applications*, Vol. 87, No. 3, pp. 163-172.

[11] Epple, M. (1998), "Topology, matter, and space: Topological notions in 19th-century natural philosophy", *Archive for history of exact sciences*, Vol. 52, No. 4, pp. 297- 392.

[12] Greenberg, M. J. (2018), *"Algebraic topology: a first course"*, CRC Press. Massey, W. S. (2019), *"A basic course in algebraic topology"*, Vol. 127. Springer.

[13] Kris Wysocki, Hyam Rubinstein. Lecture notes on metric spaces. University of Melbourne, 2005.

[14] Micheal O. Seacoid. Undergraduate Mathematics. Metric spaces, springer 2006

[15] Morris, S. A. (1989), *"Topology without Tears"*, University of New England, England, 398pp.

[16] Sugarman, C. (2014), "Using topology to explore mathematics education reform", HMC Senior Theses, https://scholarship.claremont.e

du/hmc_theses/54/, Accessed: June 10, 2021.
[17] Sims, B.T. (1976), *"Fundamentals of topology"*, Vol. 2, Macmillan Publishing Co., Inc. New York,
[18] Sydney A. Moris. Topology without tears. January 2005
[19] Zigli, D.D., Brew, L., Obeng-Denteh, W., and Kwofie, S. (2021), "On the Application of Homeomorphism on Amoeba Proteus", *Ghana Journal of Technology*, Vol. 5, No. 2, pp. 43-47.
[20] Zigli, D.D., Henry, O., and Vincent, A. (2020), "On the Application of Metric Space on DNA Hybridisation", *International journal of Applied Science and Mathematics*, Vol. 7, No. 2, pp. 70-75.

AUTHOR'S PROFILE

First Author
David Delali Zigli, is a Lecturer in the Department of Mathematical Sciences, University of Mines and Technology, Ghana. He holds MPhil in Pure Mathematics from Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana and currently pursuing PhD in Mathematics. His research interest lies in Algebraic Topology and more specifically Fundamental Groups.

Second Author
Vivian Maanu, is a Mathematics Tutor at Akrokerri College of Education in Ghana. She has an MSc in Industrial Mathematics from Kwame Nkrumah University of Science and Technology (KNUST) in Kumasi, Ghana, and is currently pursuing MPhils in Applied Mathematics and Mathematics Education. Her research interests include: modelling with differential equations, advanced instructional strategies in mathematics education, and research design and methods in mathematics education.

Third Author
Samuel Amoako, is a Senior Tutor in the Department of Mathematics at Akrokerri College of Education. He has a Master of Science degree in Industrial Mathematics from Kwame Nkrumah University of Science and Technology (KNUST) in Kumasi, Ghana. He is currently working on his PhD in Mathematics Education at KNUST's Department of Teacher Education. His research activities include using Operation Research Methods to solve Educational Problems.

Fourth Author
Prince Nimako Gyasi, was a student (2021) in the Mathematical Sciences Department. University of Mines and Technology (UMaT), Tarkwa-Ghana and currently a research and teaching assistant. His research interest lies in Topology.