Appendix

A Prompts

A.1 Prompt Used in the Case Studies in Section 4.2:

Natural language version: *"Let $z = \frac{1+i}{\sqrt{2}}$, find $(\sum_{i=1}^{1} 2(z^{i^2})) \cdot (\sum_{i=1}^{1} 2(\frac{1}{z^{i^2}}))$. The final answer is 36."*
Translate the natural language version to an Isabelle version:

```
theorem:
  fixes z::complex
  assumes h0: "z = (Complex (1/sqrt 2) (1/sqrt 2))"
  shows "(\<Sum>k::nat=1..12. (z^(k^2)))
            * (\<Sum> k::nat=1..12. 1/(z^(k^2))) =36"
```

Natural language version: "Determine the value of ab if $\log_8 a + \log_4 b^2 = 5$ and $\log_8 b + \log_4 a^2 = 7$. The final answer is 512". Translate the natural language version to an Isabelle version:

```
theorem:
  fixes a b ::real
  assumes "(ln a) / (ln 8) + (ln (b^2)) / (ln 4) = 5"
          "(ln b) / (ln 8) + (ln (a^2)) / (ln 4) = 7"
        shows "a * b = 512"
```

B Few-shot Prompts

B.1 Prompt used to formalize `algebra` problems

Natural language version: *"Simplify $\left(\frac{4}{x}\right)^{-1}\left(\frac{3x^3}{x}\right)^2\left(\frac{1}{2x}\right)^{-3}$. The final answer is $18x^8$"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes x :: real
  assumes h0 : "x \<noteq> 0"
  shows "1/(4/x) * ((3*x^3)/x)^2 * (1/(1 / (2 * x)))^3 = 18 * x^8"
```

Natural language version: *"For integers n, let*

$$f(n) = \begin{cases} n^2 & \textit{if } n \textit{ is odd,} \\ n^2 - 4n - 1 & \textit{if } n \textit{ is even.} \end{cases}$$

Find $f(f(f(f(f(4)))))$. The final answer is 1". Translate the natural language version to an Isabelle version:

```
theorem
  fixes f :: "int \<Rightarrow> int"
  assumes "\<forall>n. odd n \<longrightarrow> f n = n^2"
    and "\<forall> n. even n \<longrightarrow> f n = n^2 - 4*n -1"
  shows "f 4 = -1"
```

Natural language version: *"The volume of a cone is given by the formula $V = \frac{1}{3}Bh$, where B is the area of the base and h is the height. The area of the base of a cone is 30 square units, and its height is 6.5 units. What is the number of cubic units in its volume? The final answer is 65"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes b h v ::real
  assumes "0 < b \<and> 0 < h \<and> 0 < v"
      and "v = 1 / 3 * (b * h)"
      and "b = 30"
      and "h = 13 / 2"
    shows "v = 65"
```

Natural language version: *"If $3a + b + c = -3, a + 3b + c = 9, a + b + 3c = 19$, then find abc. The final answer is -56"*.
Translate the natural language version to an Isabelle version:

```
theorem
  fixes a b c :: real
  assumes "3 * a + b + c = -3"
    and "a + 3 * b + c = 9"
    and "a + b + 3 * c = 19"
  shows "a * b * c = -56"
```

Natural language version: *"If $f(x) = 5x - 12$, find a value for x so that $f^{-1}(x) = f(x+1)$. The final answer is $\frac{47}{24}$"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes x :: real and \<sigma>::"real \<Rightarrow> real"
  assumes "bij \<sigma>"
    and \<sigma>:"\<forall> x. \<sigma> x = 5 * x - 12"
    and "\<sigma> (x + 1) = (inv \<sigma>) x"
  shows "x = 47 / 24"
```

Natural language version: *"What is the x-coordinate for the x-intercept of the line containing the points $(7, 4)$ and $(6, 3)$? The final answer is 3"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes a b :: real
    and f :: "real \<Rightarrow> real"
  assumes h0 : "\<And>x. f x = a * x + b"
    and h1 : "f 7 = 4"
    and h2 : "f 6 = 3"
  shows "f 3 = 0"
```

Natural language version: *"Given $2^a = 32$ and $a^b = 125$ find b^a. The final answer is 243"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes a b :: real
  assumes "2 powr a = 32"
    and "a powr b = 125"
  shows "b powr a = 243"
```

Natural language version: *"Let*

$$f(x) = \begin{cases} x^2 + 9 & \textit{if } x < -5, \\ 3x - 8 & \textit{if } x \geq -5. \end{cases}$$

If $f(x) = 10$, find the sum of all possible values of x. The final answer is 6". Translate the natural language version to an Isabelle version:

```
theorem
  fixes f :: "real \<Rightarrow> real"
  assumes "\<forall> x < -5. f x = x^2 + 5"
    and "\<forall> x \<ge> -5. f x = 3 * x -8"
  shows "(\<Sum> k \<in> (f -` {10}). k) = 6"
```

Natural language version: *"Simplify $(9 - 4i) - (-3 - 4i)$. The final answer is 12"*. Translate the natural language version to an Isabelle version:

```
theorem
  fixes q e :: complex
  assumes h0 : "q = Complex (Re 9) (Im (-4))"
    and h1 : "e = Complex (Re (-3)) (Im (-4))"
  shows "q - e = 12"
```

Natural language version: *"What is the minimum possible value for y in the equation $y = x^2 - 6x + 13$? The final answer is 4"*.
Translate the natural language version to an Isabelle version:

```
theorem
  fixes x y :: real
  assumes h0 : "y = x^2 - 6 * x + 13"
  shows "4 \<le> y"
```

B.2 Prompt used to formalize `number theory` problems

Natural language version: *"If n is a positive integer such that $2n$ has 28 positive divisors and $3n$ has 30 positive divisors, then how many positive divisors does $6n$ have? The final answer is 35".* Translate the natural language version to an Isabelle version:

```
theorem
  fixes n :: nat
  assumes "n>0"
    and "card ({k. k dvd (2*n)}) = 28"
    and "card ({k. k dvd (3*n)}) = 30"
  shows "card ({k. k dvd (6*n)}) = 35"
```

Natural language version: *"Let n be the number of integers m in the range $1 \leq m \leq 8$ such that gcd$(m, 8) = 1$. What is the remainder when 3^n is divided by 8? The final answer is 1".*
Translate the natural language version to an Isabelle version:

```
theorem
  fixes n :: nat
  assumes "n = card {k::nat. gcd k 8 = 1 \<and> 1\<le>k \<and> k < 8}"
  shows "(3^n) mod 8 = (1::nat)"
```

Natural language version: *"What is the remainder when $1 + 2 + 3 + 4 + \cdots + 9 + 10$ is divided by 9? The final answer is 1".*
Translate the natural language version to an Isabelle version:

```
theorem
  "(\<Sum> k< 11. k) mod 9 = (1::nat)"
```

Natural language version: *"Cards are numbered from 1 to 100. One card is removed and the values on the other 99 are added. The resulting sum is a multiple of 77. What number was on the card that was removed? The final answer is 45".*
Translate the natural language version to an Isabelle version:

```
theorem
  fixes x :: nat
  assumes h0 : "1 \<le> x \<and> x \<le> 100"
    and h1 : "77 dvd ((\<Sum>k::nat=0..100. k)-x)"
  shows "x=45"
```

Natural language version: *"Find $9^{-1} \pmod{100}$, as a residue modulo 100. (Give an answer between 0 and 99, inclusive.) The final answer is $89 \pmod{100}$".*
Translate the natural language version to an Isabelle version:

```
theorem
  fixes x::nat
  assumes "x < 100"
    and "x*9 mod 100 = 1"
  shows "x = 89"
```

Natural language version: "Suppose m is a two-digit positive integer such that $6^{-1} \pmod{m}$ exists and $6^{-1} \equiv 6^2 \pmod{m}$. What is m? The final answer is 43".
Translate the natural language version to an Isabelle version:

```
theorem
  fixes m x :: nat
  assumes h0 : "10 \<le> m"
    and h1 : "m \<le> 99"
    and h2 : "(6 * x) mod m = 1"
    and h3 : "(x - 6^2) mod m = 0"
  shows "m = 43"
```

Natural language version: "Find $24^{-1} \pmod{11^2}$. That is, find the residue b for which $24b \equiv 1 \pmod{11^2}$. Express your answer as an integer from 0 to $11^2 - 1$, inclusive. The final answer is 116".
Translate the natural language version to an Isabelle version:

```
theorem
  fixes b::int
  assumes "\<forall>b::int. 0\<le>b \<and> b\<le>11^2 \<and> [b * 24 = 1]
      (mod (11^2))"
  shows "b = 116"
```

Natural language version: *"Given that $p \geq 7$ is a prime number, evaluate*

$$1^{-1} \cdot 2^{-1} + 2^{-1} \cdot 3^{-1} + 3^{-1} \cdot 4^{-1} + \cdots + (p-2)^{-1} \cdot (p-1)^{-1} \pmod{p}.$$

The final answer is $2 \pmod{p}$".
Translate the natural language version to an Isabelle version:

```
theorem
  fixes p :: nat
  assumes "prime p"
  and "7 \<le> p"
shows "(\<Sum> k \<in> {1..<p-1}. (inv_mod k p)* (inv_mod (k+1) p)) = 2"
```

Natural language version: *"What is the remainder when $2000+2001+2002+2003+2004+2005+2006$ is divided by 7? The final answer is 0".* Translate the natural language version to an Isabelle version:

```
theorem
  "(2000 + 2001 + 2002 + 2003 + 2004 + 2005 + 2006) mod 7 = (0::nat)"
```

Natural language version: *"One morning each member of Angela's family drank an 8-ounce mixture of coffee with milk. The amounts of coffee and milk varied from cup to cup, but were never zero. Angela drank a quarter of the total amount of milk and a sixth of the total amount of coffee. How many people are in the family? The final answer is 5".* Translate the natural language version to an Isabelle version:

```
theorem
  fixes x y n ::nat
  assumes "x / 4 + y / 6 = (x + y) / n"
    and "n\<noteq>0"
    "x\<noteq>0" "y\<noteq>0"
  shows "n = 5"
```

B.3 Prompt used for informalization

Isabelle version:

```
theorem
  fixes z::complex
  assumes h0: "z = (Complex (1/sqrt 2) (1/sqrt 2))"
  shows "(\<Sum>k::nat=1..12. (z^(k^2)))
           * (\<Sum> k::nat=1..12. 1/(z^(k^2))) =36"
```

Translate the Isabelle version to a natural language version: *"Let* $z = \frac{1+i}{\sqrt{2}}$*, find* $(\sum_{i=1}^{1} 2(z^{i^2}))\cdot(\sum_{i=1}^{1} 2(\frac{1}{z^{i^2}}))$*. The final answer is 36."*
Isabelle version:

```
theorem
  fixes a b ::real
  assumes "(ln a) / (ln 8) + (ln (b^2)) / (ln 4) = 5"
          "(ln b) / (ln 8) + (ln (a^2)) / (ln 4) = 7"
        shows "a * b = 512"
```

Translate the Isabelle version to a natural language version: "Determine the value of ab if $\log_8 a + \log_4 b^2 = 5$ and $\log_8 b + \log_4 a^2 = 7$. The final answer is 512".

C More on case Studies

C.1 Case Study 1

Here we show the output given by PaLM model 540B. The model got the most structure correctly, but made some syntactical errors.

Question:
Natural language version: "Prove that there is no function f from the set of non-negative integers into itself such that $f(f(n)) = n + 1987$ for every n." Translate the natural language version to an Isabelle version:

PaLM 540B Output:

```
theorem
  fixes f :: "nat -> nat"
  assumes "forall n::nat. f(f(n)) = n + 1987"
  shows "false"
```

C.2 Case Study 2

We show below the output given by PaLM models of different sizes.

Question:
Natural Language version: "*When all the girls at Madeline's school line up in rows of eight, there are seven left over. If instead they line up in rows of four, how many are left over? The final answer is 3.*"
Translate the natural language version to an Isabelle version:

PaLM 540B output:

```
theorem
    fixes n::nat
    assumes "n mod 8 = 7"
    shows "n mod 4 = 3"
```

Question:
Natural Language version: "*When all the girls at Madeline's school line up in rows of eight, there are seven left over. If instead they line up in rows of four, how many are left over? The final answer is 3.*"
Translate the natural language version to an Isabelle version:

PaLM 64B output:

```
theorem
    fixes girls :: nat
    assumes "(girls / 8) + 7 = girls"
    shows "girls = 3"
```

Question:
Natural Language version: "*When all the girls at Madeline's school line up in rows of eight, there are seven left over. If instead they line up in rows of four, how many are left over? The final answer is 3.*"
Translate the natural language version to an Isabelle version:

PaLM 8B output:

```
theorem
    fixes a b ::real
    assumes "(ln a) / (ln 8) + (ln (b^2)) / (ln 4) = 7"
            "(ln b) / (ln 8) + (ln (a^2)) / (ln 4) = 3"}}
    shows "a * b = 3"
```

C.3 Case Study 3

If we include the following example in the prompt in additional to the two examples shown in Appendix B.3:

Natural language version: "What is the x-coordinate for the x-intercept of the line containing the points $(7, 4)$ and $(6, 3)$? The final answer is 3". Translate the natural language version to an Isabelle version:

```
theorem
  fixes a b :: real
    and f :: "real \<Rightarrow> real"
  assumes h0 : "\<And>x. f x = a * x + b"
    and h1 : "f 7 = 4"
    and h2 : "f 6 = 3"
  shows "f 3 = 0"
```

Then Codex was able to formalize the problem perfectly:

Question:
Natural language version: "Let f be a linear function for which $f(6) - f(2) = 12$. What is $f(12) - f(2)$? The final answer is 30."
Translate the natural language version to an Isabelle version:

Codex Output:

```
theorem
    fixes a b :: real
        and f :: "real \<Rightarrow> real"
    assumes h0 : "\<And>x. f x = a * x + b"
        and h1 : "f 6 - f 2 = 12"
    shows "f 12 - f 2 = 30"
```

Note that this additional example in the prompt is only tangentially related. The phrase “linear function” is not even mentioned. Instead, it shows an example how to formalize a *line*. But that is sufficient for the model to understand how to formalize a “linear function”.

D More Examples on Mathematical Competition Problems

D.1 Perfect Translations

Question:
Natural language version: *"At what point does the line containing the points* $(1, 7)$ *and* $(3, 11)$ *intersect the* y*-axis? Express your answer as an ordered pair. The final answer is (0,5)".*
Translate the natural language version to an Isabelle version:

Codex Output:

```
theorem
  fixes a b :: real
    and f :: "real \<Rightarrow> real"
  assumes h0 : "\<And>x. f x = a * x + b"
    and h1 : "f 1 = 7"
    and h2 : "f 3 = 11"
  shows "f 0 = 5"
```

Question:
Natural language version: *"A map of the town that Annie, Barbara, and Charlie live in can be represented by the Cartesian plane. Annie is located at* $(6, -20)$ *and Barbara is located at* $(1, 14)$*. They agree to meet at the closest point that is equidistant from their current locations and walk upwards together to get to Charlie's location at* $\left(\frac{7}{2}, 2\right)$*. How many units upward do Annie and Barbara walk together to get to Charlie? The final answer is 5."* Translate the natural language version to an Isabelle version:

Codex Output:

```
theorem
  fixes a b c :: "real \<times> real"
  assumes h0 : "a = (6, -20)"
    and h1 : "b = (1, 14)"
    and h2 : "c = (7 / 2, 2)"
  shows "dist a c + dist b c = 5"
```

Question:
Natural language version: "Let x be a positive integer, and define the integers $n = x^2+2x+17$ and $d = 2x + 5$. When dividing n by d, the quotient is x, and the remainder is 7. Find x. The final answer is 2".

Codex Output:

```
theorem
  fixes x n d :: int
  assumes h0 : "x > 0"
    and h1 : "n = x^2 + 2 * x + 17"
    and h2 : "d = 2 * x + 5"
    and h3 : "n div d = x"
    and h4 : "n mod d = 7"
  shows "x = 2"
```

Question:
Natural language version: "A line segment begins at $(1, 3)$. It is 5 units long and ends at the point $(4, x)$ and $x > 0$. What is the value of x? The final answer is 7".

Codex Output:

```
theorem
  fixes x :: real
  assumes h0 : "x > 0"
    and h1 : "5 = sqrt ((4 - 1)^2 + (x - 3)^2)"
  shows "x = 7"
```

Natural language version: "A line segment begins at $(1, 3)$. It is 5 units long and ends at the point $(4, x)$ and $x > 0$. What is the value of x? The final answer is 7".

Codex Output:

```
theorem
  fixes x :: real
  assumes h0 : "x > 0"
    and h1 : "5 = sqrt ((4 - 1)^2 + (x - 3)^2)"
  shows "x = 7"
```

Question:
Natural language version: "Let $f(x) = x^2 - 2x$. How many distinct real numbers c satisfy $f(f(f(f(c)))) = 3$? The final answer is 9".

Codex Output:

```
theorem
  fixes f :: "real \<Rightarrow> real"
  assumes h0 : "\<forall> x. f x = x^2 - 2 * x"
  shows "card {c. f (f (f (f c))) = 3} = 9"
```

Question:
Natural language version: "Find the solutions to $z^4 = -4$. Enter the solutions, separated by commas. The final answer is 1 + i, 1 - i, -1 + i, -1 - i".

Codex Output:

```
theorem
  fixes z :: complex
  assumes "z^4 = -4"
  shows "z = 1 + \<i> \<or> z = 1 - \<i> \<or> z = -1 + \<i> \<or> z = -1 -
      \<i>"
```

Question:
Natural language version: "Let z be a complex number satisfying $z^2 = 4z - 19 + 8i$. Given that $|z|$ is an integer, find z. The final answer is 3 + 4i".

Codex Output:

```
theorem
  fixes z :: complex
  assumes h0 : "z^2 = 4 * z - 19 + 8 * \<i>"
    and h1 : "norm z \<in> \<int>"
  shows "z = 3 + 4 * \<i>"
```

Question:
Natural language version: "The sequence (a_n) satisfies

$$a_1 + a_2 + a_3 + \cdots + a_n = n^2 a_n$$

for all $n \geq 2$. If $a_{63} = 1$, find a_1. The final answer is 2016".

Codex Output:

```
theorem
  fixes a :: "nat \<Rightarrow> real"
  assumes h0 : "\<forall> n \<ge> 2. (\<Sum> k = 1..n. a k) = n^2 * a n"
    and h1 : "a 63 = 1"
  shows "a 1 = 2016"
```

Question:
Natural language version: "Find the units digit of the following within the indicated number base: $52_7 + 62_7$ The final answer is 4".

Codex Output:

```
theorem
  fixes x y :: nat
  assumes "x < 7"
    and "y < 7"
  shows "((x*7+2) + (y*7+2)) mod 7 = 4"
```

Question:
Natural language version: "How many factors of 8000 are perfect squares? The final answer is 8".

Codex Output:

```
theorem
  fixes n :: nat
  assumes "n = card {k::nat. k^2 dvd 8000}"
  shows "n = 8"
```

Question:
Natural language version: "Solve the congruence $11n \equiv 7 \pmod{43}$, as a residue modulo 43. (Give an answer between 0 and 42.) The final answer is 28".

Codex Output:

```
theorem
  fixes n :: nat
  assumes "n < 43"
    and "(11 * n) mod 43 = 7"
  shows "n = 28"
```

Question:
Natural language version: "What is the 100th digit after the decimal point of the decimal representation of 1/7? The final answer is 8".

Codex Output:

```
theorem
  fixes x :: nat
  assumes "x = (1/7) * 10^100"
  shows "x mod 10 = 8"
```

Question:
Natural language version: "Two positive integers m and n are chosen such that m is the smallest positive integer with only two positive divisors and n is the largest integer less than 100 with exactly three positive divisors. What is $m + n$? The final answer is 51".

Codex Output:

```
  theorem
  fixes m n :: nat
  assumes "m = (LEAST k. card {d. d dvd k \<and> d > 0} = 2)"
    and "n = (GREATEST k. card {d. d dvd k \<and> d > 0} = 3 \<and> k < 100)"
  shows "m + n = 51"
```

D.2 Incorrect Translations

In this section, we present failure cases for each of the categories in Table 2 in Section. 4.4.

Error category: incomplete/ill-formed/unclear prompt
Question:
Natural language version: "Find the largest value of t such that

$$\frac{13t^2 - 34t + 12}{3t - 2} + 5t = 6t - 1.$$

The final answer is \frac{5".

Codex Output:

```
theorem
  fixes t :: real
  assumes h0 : "13 * t^2 - 34 * t + 12 = 3 * t - 2"
    and h1 : "5 * t = 6 * t - 1"
  shows "t = 5 / 2"
```

Error category: fail to align definitions or concepts
Question:
Natural language version: "What is the smallest four-digit positive integer that is divisible by 47? The final answer is 1034".

Codex Output:

```
theorem
  fixes x :: nat
  assumes "x > 999"
    and "x < 10000"
    and "47 dvd x"
  shows "x = 1034"
```

Here, the translator correctly captured the meaning of "four-digit positive integer" (i.e., `"x > 999"` and `"x < 10000"`) and that of "divisible by 47" (i.e., `"47 dvd x"`). However, it fails to align the "smallest" concept to the one in Isabelle.

Error category: inconsistent/missing assumption
Question:
Natural language version: "Jack walked up a hill at a speed of $(x^2 - 11x - 22)$ miles per hour. Meanwhile, Jill walked a total distance of $(x^2 - 3x - 54)$ miles in $(x + 6)$ hours. If Jack and Jill walked at the same speed, what is that speed, in miles per hour? The final answer is 4".

Codex Output:

```
theorem
  fixes x :: real
  assumes h0 : "x^2 - 11 * x - 22 = x^2 - 3 * x - 54"
  shows "x = 4"
```

Here, the translator missed the "$(x + 6)$" part in the natural language—it should be translated to `"x^2 - 11 * x - 22 = (x^2 - 3 * x - 54)/(x+6)"` instead. Also, the conclusion is incorrect either: it should be `"x^2 - 11 * x - 22 = 4"` or `"(x^2 - 3 * x - 54)/(x+6)= 4"`.

Error category: syntactical/type error)
Question:
Natural language version: "Find

$$\sum_{N=1}^{1024} \lfloor \log_2 N \rfloor.$$

The final answer is 8204".

Codex Output:

```
theorem
  fixes N :: nat
  shows "(\<Sum> N = 1..1024. nat \<lfloor>log 2 N\<rfloor>) = 8204"
```

In the current version, `N` will be inferred as a real number, so that the interval "`1..1024`" will be a real interval from 1 to 1024. This is incorrect, and sum over such infinite interval will result in 0 (i.e., `"(\<Sum> N = 1..1024. nat \<lfloor>log 2 N\<rfloor>)= 0"`). The correct version requires an explicit type conversion: `"(\<Sum> N = 1..1024. \<lfloor>log 2 (of_int N)\<rfloor>)= 8204"`.

Error category: missing definition in Isabelle
Question:
Natural language version: "The product of two positive three-digit palindromes is 436,995. What is their sum? The final answer is 1332".

Codex Output:

```
theorem
  fixes x y :: nat
  assumes "100 \<le> x \<and> x \<le> 999"
    and "100 \<le> y \<and> y \<le> 999"
    and "x * y = 436995"
  shows "x + y = 1332"
```

The only missing part in this example is the definition of palindromes. It appears that the translator does not know what a palindrome is so that it chooses to drop it from the assumptions.

Error category: wrong application of functions
Question:
Natural language version: "If the operation Z is defined as $aZb = b + 10a - a^2$, what is the value of $2Z6$? The final answer is 22".

Codex Output:

```
theorem
  fixes a b :: real
  assumes h0 : "a Z b = b + 10 * a - a^2"
  shows "2 Z 6 = 22"
```

The function `Z` should be applied in a prefix way in Isabelle. Also, the translator missed the "for all" quantifier in the assumption. The correct assumption should be `"\<forall> a b. Z a b = b + 10 * a - a^2"`.

Error category: other
Question:
Natural language version: "Simplify: $|-3^2 + 4|$ The final answer is 5".

Codex Output:

```
theorem
  fixes x :: real
  assumes h0 : "x = - 3 ^2 + 4"
  shows "abs x = 5"
```

Here, a parenthesis should be placed around 3 ^2 (i.e., "x = - (3 ^2)+ 4"). This is a subtle issue to operator precedence in Isabelle.

Error category: other
Question:
Natural language version: "How many numbers from 1 to 150 are not perfect squares or perfect cubes? The final answer is 135".

Codex Output:

```
theorem
  fixes n :: nat
  assumes "n \<le> 150"
  shows "card {k::nat. k \<le> n \<and> \<not> (\<exists> m. m^2 = k \<or>
      m^3 = k) } = 135"
```

The translator somehow introduces an unnecessary variable n and assumed "n \<le> 150". The correct translation should have n replaced with 150 in the conclusion: "card {k::nat. k \<le> 150 \<and> \<not> (\<exists> m. m^2 = k \<or> m^3 = k)} = 135".

E More Examples on Advanced Mathematics

E.1 Formalization

Question:
Natural language version: "Let f be a continuous function from a compact convex set S to itself. Then there exists $x \in S$ such that $f(x) = x$." Translate the natural language version to an Isabelle version:

Codex Output:

```
theorem
  fixes f :: "'a::real_normed_vector \<Rightarrow> 'a"
  assumes "continuous_on S f"
          "compact S"
          "convex S"
        shows "\<exists>x\<in>S. f x = x"
```

Here, we asked the model to formalize the statement of the famous Brouwer's fixed-point theorem. The model made two mistakes. The first one is that the model erroneously assume the function is on a real normed vector space rather than a euclidean space or a banach space. This is quite understandable as we left this information in the source. The second and more critical error is that the absence of the assumption of f mapping from S to itself (i.e., $f(S) \subseteq S$ or $f(S) = S$).

Question: Let γ be a closed path in the complex plane, and z be a point not on γ. Then the winding number of γ around z is an integer.

Codex Output:

```
  theorem
  fixes \<gamma> :: "real \<Rightarrow> complex" and z :: complex
  assumes "continuous_on {0..1} \<gamma>"
          "\<gamma> 0 = \<gamma> 1"
          "z \<notin> \<gamma> ` {0..1}"
        shows "winding_number \<gamma> z \<in> \<int>"
```

Here, the model has made a perfect translation–not only the model correctly infers that a path in the complex plane should be a function from the real interval $[0, 1]$ to complex, it also manages to align a path being closed to the assumption `\<gamma> 0 = \<gamma> 1`.

E.2 Informalization

The following lemma statements are from the standard library of Isabellle (i.e., HOL-Complex_Analysis and HOL-Probability).

proposition *openin_clauses:*
  fixes *U :: "'a topology"*
  shows
    "openin U {}"
    "$\bigwedge$S T. openin U S $\Longrightarrow$ openin U T $\Longrightarrow$ openin U (S$\cap$T)"
    "$\bigwedge$K. ($\forall$S $\in$ K. openin U S) $\Longrightarrow$ openin U ($\bigcup$K)"

Translate the Isabelle version to a natural language version:

Codex Output:

```
Let $U$ be a topology, then $U$ is open if and only if it satisfies the
    following conditions:
1. $U$ is open.
2. If $S$ and $T$ are open, then $S \cap T$ is open.
3. If $K$ is a collection of open sets, then $\cup K$ is open.
```

lemma *openin_Int[intro]:*
    "openin U S $\Longrightarrow$ openin U T
      $\Longrightarrow$ openin U (S $\cap$ T)"

Translate the Isabelle version to a natural language version:

Codex Output:
"Let U be a topology, then $\emptyset$ is open in U, if S and T are open in U, then $S \cap T$ is open in U, if K is a collection of sets, and each set in K is open in U, then $\bigcup K$ is open in U".

lemma *openin_Int[intro]:*
    "openin U S $\Longrightarrow$ openin U T
      $\Longrightarrow$ openin U (S $\cap$ T)"

Translate the Isabelle version to a natural language version:

Codex Output:
"If S and T are open in U, then $S \cap T$ is open in U".

lemma *openin_Inter [intro]:*
  assumes *"finite $\mathcal{F}$" "$\mathcal{F} \neq$ {}" "$\bigwedge$X. X $\in \mathcal{F}$ $\Longrightarrow$ openin T X"*
  shows *"openin T ($\bigcap\mathcal{F}$)"*

Translate the Isabelle version to a natural language version:

Codex Output:
"If T is a topological space, and $\mathcal{F}$ is a finite non-empty collection of open subsets of T, then $\bigcap \mathcal{F}$ is open in T".

lemma *openin_subtopology:*

```
"openin (subtopology U V) S ⟷ (∃T. openin U T ∧ S = T ∩ V)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"S is open in the subtopology of U induced by V if and only if there exists an open set T in U such that $S = T \cap V$".

lemma *closedin_subtopology_Un:*

```
"⟦closedin (subtopology X T) S; closedin (subtopology X U) S⟧
 ⟹ closedin (subtopology X (T ∪ U)) S"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If S is closed in T and S is closed in U, then S is closed in $T \cup U$".

lemma *connected_closedin:*

```
"connected S ⟷
 (E1 E2.
  closedin (top_of_set S) E1 ∧
  closedin (top_of_set S) E2 ∧
  S ⊆ E1 ∪ E2 ∧ E1 ∩ E2 = {} ∧ E1 ≠ {} ∧ E2 ≠ {})"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"A set S is connected if and only if there does not exist two closed sets E_1 and E_2 such that $S \subseteq E_1 \cup E_2$, $E_1 \cap E_2 = \emptyset$, $E_1 \neq \emptyset$ and $E_2 \neq \emptyset$".

definition *continuous_map* **where**

```
"continuous_map X Y f ≡
   (∀x ∈ topspace X. f x ∈ topspace Y) ∧
   (∀U. openin Y U ⟶ openin X {x ∈ topspace X. f x ∈ U})"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"A function f from a topological space X to a topological space Y is continuous if for every open set U in Y, the preimage $f^{-1}(U)$ is open in X".

definition *continuous_map* **where**

```
"continuous_map X Y f ≡
   (∀x ∈ topspace X. f x ∈ topspace Y) ∧
   (∀U. openin Y U ⟶ openin X {x ∈ topspace X. f x ∈ U})"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"A continuous map $f : X \to Y$ is a map such that for every open set $U \subset Y$, the preimage $f^{-1}(U)$ is open in X".

```
lemma connectedin_separation:
  "connectedin X S ⟷
        S ⊆ topspace X ∧
        (∄C1 C2. C1 ∪ C2 = S ∧ C1 ≠ {} ∧ C2 ≠ {}
        ∧ C1 ∩ X closure_of C2 = {} ∧ C2
          ∩ X closure_of C1 = {})" (is "?lhs = ?rhs")
```

Translate the Isabelle version to a natural language version:

Codex Output:
"S is connected in X if and only if $S \subseteq X$ and there do not exist two non-empty sets C_1 and C_2 such that $C_1 \cup C_2 = S$ and $C_1 \cap X\overline{C_2} = \emptyset$ and $C_2 \cap X\overline{C_1} = \emptyset$".

```
lemma connectedin_Union:
  assumes 𝒰: "⋀S. S ∈ 𝒰 ⟹ connectedin X S" and ne: "⋂𝒰 ≠ {}"
  shows "connectedin X (⋃𝒰)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If S is connected in X for all $S \in \mathcal{U}$ and $\bigcap_{S\in\mathcal{U}} S \neq \emptyset$, then $\bigcup_{S\in\mathcal{U}} S$ is connected in X".

```
lemma compact_space_alt:
   "compact_space X ⟷
        (∀𝒰. (∀U ∈ 𝒰. openin X U) ∧ topspace X ⊆ ⋃𝒰
            ⟶ (∃ℱ. finite ℱ ∧ ℱ ⊆ 𝒰 ∧ topspace X ⊆ ⋃ℱ))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"A topological space X is compact if and only if for every open cover $\mathcal{U}$ of X, there exists a finite subcover $\mathcal{F}$ of $\mathcal{U}$".

```
lemma compactin_subtopology:
      "compactin (subtopology X S) T ⟷ compactin X T ∧ T ⊆ S"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"T is compact in S if and only if T is compact in X and $T \subseteq S$".

```
lemma closedin_compact_space:
   "⟦compact_space X; closedin X S⟧ ⟹ compactin X S"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If X is a compact space and S is a closed subset of X, then S is a compact space".

```
lemma compactin_discrete_topology:
  "compactin (discrete_topology X) S ⟷ S ⊆ X ∧ finite S"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"S is compact in the discrete topology on X if and only if S is a finite subset of X".

```
lemma image_compactin:
  assumes cpt: "compactin X S" and cont: "continuous_map X Y f"
  shows "compactin Y (f ` S)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If S is compact in X and f is continuous from X to Y, then $f(S)$ is compact in Y".

```
lemma continuous_on_open:
  "continuous_on S f ⟷
    (∀T. openin (top_of_set (f ` S)) T ⟶
      openin (top_of_set S) (S ∩ f -` T))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"f is continuous on S if and only if for any open set T in $f(S)$, $f^{-1}(T)$ is open in S".

```
lemma continuous_closedin_preimage_gen:
  assumes "continuous_on S f" "f ` S ⊆ T"
          "closedin (top_of_set T) U"
  shows "closedin (top_of_set S) (S ∩ f -` U)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If f is continuous on S and $f(S) \subseteq T$, and U is closed in T, then $S \cap f^{-1}(U)$ is closed in S".

```
theorem Casorati_Weierstrass:
  assumes "open M" "z ∈ M" "f holomorphic_on (M - {z})"
      and "⋀l. ¬ (f ⤏ l) (at z)"
          "⋀l. ¬ ((inverse ∘ f) ⤏ l) (at z)"
    shows "closure(f ` (M - {z})) = UNIV"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a holomorphic function on an open set M containing z. If f is not constant in any neighborhood of z and f^{-1} is not constant in any neighborhood of z, then $f(M - \{z\})$ is dense in $\mathbb{C}$".

```
theorem
  assumes "open M" "z ∈ M" "f holomorphic_on (M - {z})"
      and "⋀l. ¬ (f  l) (at z)"
          "⋀l. ¬ ((inverse ∘ f)  l) (at z)"
    shows "closure(f ` (M - {z})) = UNIV"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let M be an open set and $z \in M$. Let f be a holomorphic function on $M - \{z\}$. If f is not bounded in any neighborhood of z and $\frac{1}{f}$ is not bounded in any neighborhood of z, then $f(M - \{z\})$ is dense in $\mathbb{C}$.

```
theorem great_Picard:
  assumes "open M" "z ∈ M" "a ≠ b"
                        and holf: "f holomorphic_on (M - {z})"
      and fab: "⋀w. w ∈ M - {z} ⟹ f w ≠ a ∧ f w ≠ b"
    obtains l where "(f  l) (at z)
      ∨ ((inverse ∘ f)  l) (at z)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a holomorphic function on $M - \{z\}$, where M is an open set and $z \in M$. If $f(z) \neq a$ and $f(z) \neq b$, then f or $\frac{1}{f}$ has a limit at z".

```
theorem brouwer:
  fixes f :: "'a::euclidean_space ⇒ 'a"
  assumes S: "compact S" "convex S" "S ≠ {}"
    and contf: "continuous_on S f"
    and fim: "f ` S ⊆ S"
  obtains x where "x ∈ S" and "f x = x"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a continuous function from a compact convex set S to itself. Then there exists $x \in S$ such that $f(x) = x$".

```
theorem Jordan_curve:
  fixes c :: "real ⇒ complex"
  assumes "simple_path c" and loop: "pathfinish c = pathstart c"
  obtains inner outer where
                "inner ≠ {}" "open inner" "connected inner"
                "outer ≠ {}" "open outer" "connected outer"
                "bounded inner" "¬ bounded outer" "inner ∩ outer = {}"
                "inner ∪ outer = - path_image c"
                "frontier inner = path_image c"
                "frontier outer = path_image c"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let c be a simple closed curve. Then there exist two disjoint open sets U and V such that U is bounded, V is unbounded, $U \cup V = \mathbb{C} \setminus c$, $U \cap V = \emptyset$, $\partial U = c$, and $\partial V = c$".

```
corollary Jordan_disconnected:
  fixes c :: "real ⇒ complex"
  assumes "simple_path c" "pathfinish c = pathstart c"
  shows "¬ connected(- path_image c)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If c is a simple closed curve, then $R^2 - c$ is not connected".

```
theorem inverse_squares_sums: "(λn. 1 / (n + 1)²) sums (pi² / 6)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Find the sum of the series $\sum_{i=1}^{\infty} \frac{1}{(i+1)^2}$. The final answer is $\frac{\pi^2}{6}$".

```
proposition connected_space_product_topology:
   "connected_space(product_topology X I) ⟷
    (Π_E i∈I. topspace (X i)) = {} ∨ (∀i ∈ I. connected_space(X i))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"The product topology of a family of topological spaces is connected if and only if the product of the topological spaces is empty or each topological space is connected".

```
theorem (in ring_of_sets) caratheodory':
  assumes posf: "positive M f" and ca: "countably_additive M f"
  shows "∃μ :: 'a set ⇒ ennreal.
      (∀s ∈ M. μ s = f s) ∧ measure_space Ω (sigma_sets Ω M) μ"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let M be a σ-algebra on Ω, and f be a positive countably additive function on M. Then there exists a measure μ on M such that $\mu(s) = f(s)$ for all $s \in M$ and (Ω, M, μ) is a measure space".

```
lemma sigma_finite_density_unique:
  assumes borel: "f ∈ borel_measurable M" "g ∈ borel_measurable M"
  and fin: "sigma_finite_measure (density M f)"
shows "density M f = density M g ⟷ (AE x in M. f x = g x)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f and g be two measurable functions on a sigma-finite measure space M. Then f and g are equal almost everywhere if and only if the measures μ_f and μ_g are equal".

```
lemma mvt_simple:
  fixes f :: "real ⇒ real"
  assumes "a < b"
    and derf: "⋀x. ⟦a ≤ x; x ≤ b⟧
                ⟹ (f has_derivative f' x) (at x within {a..b})"
  shows "∃x∈{a<..<b}. f b - f a = f' x (b - a)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a function defined on $[a, b]$ and differentiable on (a, b). Then there exists a point $c \in (a, b)$ such that $f(b) - f(a) = f'(c)(b - a)$".

```
theorem second_countable_borel_measurable:
  fixes X :: "'a::second_countable_topology set set"
  assumes eq: "open = generate_topology X"
  shows "borel = sigma UNIV X"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let X be a set of open sets in a second countable topology. Then the Borel σ-algebra is the σ-algebra generated by X".

```
theorem borel_measurable_ereal_iff_real:
  fixes f :: "'a ⇒ ereal"
  shows "f ∈ borel_measurable M ⟷
    ((λx. real_of_ereal (f x)) ∈ borel_measurable M ∧ f -` {∞}
              ∩ space M ∈ sets M ∧ f -` {-∞} ∩ space M ∈ sets M)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a function from M to $\mathbb{R}$, then f is measurable if and only if f is measurable and $f^{-1}(\infty)$ and $f^{-1}(-\infty)$ are measurable sets".

```
theorem condensation_test:
  assumes mono: "⋀m. 0 < m ⟹ f (Suc m) ≤ f m"
  assumes nonneg: "⋀n. f n ≥ 0"
  shows "summable f ⟷ summable (λn. 2^n * f (2^n))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a non-negative function such that $f(n + 1) \leq f(n)$ for all n. Prove that f is summable if and only if $2^n f(2^n)$ is summable".

```
theorem ratio_test_convergence:
  fixes f :: "nat ⇒ real"
  assumes pos_f: "eventually (λn. f n > 0) sequentially"
  defines "l ≡ liminf (λn. ereal (f n / f (Suc n)))"
  assumes l: "l > 1"
  shows   "summable f"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a positive function, and let l be the limit inferior of the sequence $\frac{f(n)}{f(n+1)}$. If $l > 1$, then f is summable".

```
theorem Stone_Weierstrass_polynomial_function:
  fixes f :: "'a::euclidean_space ⇒ 'b::euclidean_space"
  assumes S: "compact S"
      and f: "continuous_on S f"
      and e: "0 < e"
    shows "∃g. polynomial_function g ∧ (∀x ∈ S. norm(f x - g x) < e)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a continuous function on a compact set S. Then there exists a polynomial function g such that $\forall x \in S, |f(x) - g(x)| < \epsilon$.

```
theorem Riemann_mapping_theorem:
    "open S ∧ simply_connected S ⟷
     S = {} ∨ S = UNIV ∨
     (∃f g. f holomorphic_on S ∧ g holomorphic_on ball 0 1 ∧
          (∀z ∈ S. f z ∈ ball 0 1 ∧ g(f z) = z) ∧
          (∀z ∈ ball 0 1. g z ∈ S ∧ f(g z) = z))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let S be a simply connected open set. Then S is either empty, the whole plane, or there exists a holomorphic function f from S to the unit disk and a holomorphic function g from the unit disk to S such that $g \circ f$ is the identity function on S and $f \circ g$ is the identity function on the unit disk."

```
theorem Residue_theorem:
  fixes s pts::"complex set" and f::"complex ⇒ complex"
    and g::"real ⇒ complex"
  assumes "open s" "connected s" "finite pts" and
          holo:"f holomorphic_on s-pts" and
          "valid_path g" and
          loop:"pathfinish g = pathstart g" and
          "path_image g ⊆ s-pts" and
          homo:"∀z. (z ∉ s) ⟶ winding_number g z  = 0"
        shows "contour_integral g f =
          2 * pi * i *(∑p∈pts. winding_number g p * residue f p)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let f be a holomorphic function on a connected open set S and let g be a closed path in S that does not pass through any of the points in the finite set P. Then $\int_g f(z)dz = 2\pi i \sum_{p \in P} Res(f, p)$".

```
theorem integer_winding_number:
  "⟦path γ; pathfinish γ = pathstart γ;
  z ∉ path_image γ⟧ ⟹ winding_number
  γ z ∈ ℤ"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let γ be a closed path in the complex plane, and z be a point not on γ. Then the winding number of γ around z is an integer".

```
lemma winding_number_eq:
     "⟦path γ; pathfinish γ
        = pathstart γ; w ∈ S; z ∈ S; connected S;
        S ∩ path_image γ = {}⟧
      ⟹ winding_number γ w = winding_number γ z"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let γ be a closed path in the plane, and let w, z be two points in the plane. If γ does not pass through w or z, and w, z are in the same connected component of the plane, then the winding number of γ about w is the same as the winding number of γ about z".

```
corollary winding_number_zero_outside:
    "⟦path γ; convex s; pathfinish γ
    = pathstart γ; z ∉ s; path_image γ
    ⊆ s⟧ ⟹ winding_number γ z = 0"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If γ is a path, s is a convex set, γ is a closed path, z is not in s, and γ is in s, then the winding number of γ around z is 0".

```
lemma winding_number_zero_at_infinity:
  assumes γ: "path γ" and loop: "pathfinish γ = pathstart γ"
  shows "∃B. ∀z. B ≤ norm z ⟶ winding_number γ z = 0"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let γ be a closed path, then there exists a constant B such that for all z, if $|z| > B$, then the winding number of γ at z is 0".

```
lemma winding_number_homotopic_paths:
    assumes "homotopic_paths (-{z}) g h"
    shows "winding_number g z = winding_number h z"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If g and h are homotopic paths in $\mathbb{C} - \{z\}$, then the winding number of g around z is equal to the winding number of h around z".

```
lemma simple_closed_path_winding_number_cases:
  assumes "simple_path γ" "pathfinish γ = pathstart γ"
          "z ∉ path_image γ"
  shows "winding_number γ z ∈ {-1,0,1}"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If γ is a simple closed path, z is not in the image of γ, then the winding number of γ at z is either -1, 0 or 1".

```
corollary Cauchy_theorem_primitive:
  assumes "⋀x. x ∈ S ⟹
            (f has_field_derivative f' x) (at x within S)"
      and "valid_path g" "path_image g ⊆ S"
          "pathfinish g = pathstart g"
    shows "(f' has_contour_integral 0) g"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If f is a function with a derivative f' on a set S, and g is a closed path in S, then $\int_g f'(z)dz = 0$".

```
theorem (in prob_space) central_limit_theorem_zero_mean:
  fixes X :: "nat ⇒ 'a ⇒ real"
    and μ :: "real measure"
    and σ :: real
    and S :: "nat ⇒ 'a ⇒ real"
  assumes X_indep: "indep_vars (λi. borel) X UNIV"
    and X_mean_0: "⋀n. expectation (X n) = 0"
    and σ_pos: "σ > 0"
    and X_square_integrable: "⋀n. integrable M (λx. (X n x)²)"
    and X_variance: "⋀n. variance (X n) = σ²"
    and X_distrib: "⋀n. distr M borel (X n) = μ"
  defines "S n ≡ λx. ∑i<n. X i x"
  shows "weak_conv_m (λn. distr M borel
          (λx. S n x / sqrt (n * σ²))) std_normal_distribution"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"Let $X_1, X_2, \cdots$ be a sequence of independent random variables with mean 0 and variance σ^2. Let $S_n = \sum_{i=1}^{n} X_i$. Then $S_n/\sqrt{n\sigma^2}$ converges in distribution to a standard normal distribution".

```
proposition compact_space_product_topology:
   "compact_space(product_topology X I) ⟷
    topspace(product_topology X I) = {} ∨ (∀i ∈ I. compact_space(X i))"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If X_i is a compact space for all $i \in I$, then $X_1 \times X_2 \times \cdots \times X_n$ is a compact space".

Lemma statements in the remainder of this section are from the *Topology* entry in the AFP.[3] All the natural language version is generated by the Codex model.

```
lemma seteqI:
  "⟦⋀x. x∈A ⟹ x∈B; ⋀x. x∈B ⟹ x∈A ⟧ ⟹ A = B"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If A and B are sets such that $A \subseteq B$ and $B \subseteq A$, then $A = B$."

```
lemma (in carrier) openI: "m ∈ T ⟹ m open"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If m is a topology on a set T, then m is open."

[3] https://www.isa-afp.org/browser_info/current/AFP/Topology/Topology.html

```
lemma (in carrier) openE: "⟦ m open; m ∈ T ⟹ R ⟧ ⟹ R"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If m is an open set in a topological space T, then $m \in T$."

```
lemma (in topology) open_kriterion:
  assumes t_contains_open: "⋀ x. x∈t ⟹ ∃t'. t' open ∧ x∈t' ∧ t'⊆t"
  shows "t open"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If t is a subset of a topological space T such that for every $x \in t$, there exists an open set t' such that $x \in t'$ and $t' \subseteq t$, then t is open."

```
lemma (in carrier) carrierI [intro]:
  "⟦ t open; x ∈ t ⟧ ⟹ x ∈ carrier"
```

Translate the Isabelle version to a natural language version: "If t is an open set in a topological space T and $x \in t$, then $x \in T$."

```
lemma (in topology) Un_open [intro!]:
  assumes abopen: "A open" "B open"
  shows "A ∪ B open"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If A and B are open sets in a topological space T, then $A \cup B$ is open."

```
lemma topologyI:
  "⟦⋀ x y. ⟦ is_open T x; is_open T y⟧ ⟹ is_open T (x ∩ y);
    ⋀ M. ∀ m ∈ M. is_open T m ⟹ is_open T (⋃ M)
  ⟧ ⟹ topology T"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If T is a set and T is closed under finite intersections and arbitrary unions, then T is a topology."

```
lemma (in topobase) topo_open:
  "t open = (t ∈ topo B)"
```

Translate the Isabelle version to a natural language version:

Codex Output:
"If B is a topological base of a topological space T, then t is open if and only if $t \in B$."

```
lemma (in topobase) topo_induct
  [case_names basic inter union, induct set: topo, consumes 1]:
  assumes opn: "x open"
    and bas: "⋀x. x ∈ B ⟹ P x"
    and int: "⋀x y. ⟦x open; P x; y open; P y⟧ ⟹ P (x ∩ y)"
    and uni: "⋀M. (∀t∈M. t open ∧ P t) ⟹ P (⋃M)"
  shows "P x"
```

Translate the Isabelle version to an natural language version:

Codex Output:
"If B is a topological base of a topological space T, then $P(x)$ holds for all open sets x if $P(x)$ holds for all $x \in B$ and $P(x)$ is closed under finite intersections and arbitrary unions."

```
lemma subtopologyI:
  fixes S (structure)
  fixes T (structure)
  assumes H1: "⋀s. s open ⟹ ∃t. t open_T ∧ s = t ∩ carrier"
    and   H2: "⋀t. t open_T ⟹ t ∩ carrier open"
  shows "subtopology S T"
```

Translate the Isabelle version to an natural language version:

Codex Output:
"If S is a subset of a topological space T such that for every open set s in S, there exists an open set t in T such that $s = t \cap S$ and for every open set t in T, $t \cap S$ is open in S, then S is a subspace of T."

```
lemma (in subtopology) subtopologyE [elim]:
  assumes major: "s open"
    and   minor: "⋀t. ⟦ t open_T; s = t ∩ carrier ⟧ ⟹ R"
  shows "R"
```

Translate the Isabelle version to an natural language version:

Codex Output:
"If s is an open set in a topological space T, then s is an open set in a subspace S of T if and only if $s = t \cap S$ for some open set t in T."

```
lemma continuousI:
  fixes S (structure)
  fixes T (structure)
  assumes "f : carrier_S → carrier_T"
          "⋀m. m open_T ⟹ carrier ∩ (f -` m) open"
  shows "continuous f S T"
```

Translate the Isabelle version to an natural language version:

Codex Output:
"If f is a function from a topological space S to a topological space T such that the preimage of every open set in T is open in S, then f is continuous."

```
lemma continuousE:
  fixes S (structure)
  fixes T (structure)
  shows
    "⟦continuous f S T;
       ⟦f : carrier_S → carrier_T;
        ∀m. m open_T ⟶ carrier_S ∩ (f -` m) open ⟧ ⟹ P
    ⟧ ⟹ P"
```

Translate the Isabelle version to an natural language version:

Codex Output:

"If x is a limit point of a set A in a topological space T, then there exists a filter F such that $x \in F$ and $A \in F$."