The Title of My Paper

Wolfgang Schreiner
Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, Linz, Austria
Wolfgang.Schreiner@risc.jku.at

March 15, 2021

Abstract

This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract. This is my not really very long abstract.

1 Introduction

Note these beautiful Unicode symbols: ä ö ü € and ß. See my papers [6, 5]. Also see [1] and [3] and do not overlook [2] and [4]. See Section 5.

This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph. This is another paragraph.

The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation. The typographically very important effect of hyphenation.

Figure 1: My Figure

2 My Results

See Figure 1 on page 2.

One Thought   Some formulas:

$$(a + b)^2 = a^2 + 2ab + b^2$$
$$SUM = SUM = S \cdot U \cdot M$$

No new paragraph (no indentation).

Another Thought   The Gauss formula $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ displayed:

$$\sum_{i=1}^{n} i = \frac{n(n + 1)}{2}$$

Now a new paragraph (indentation): Einstein says $E = mc^2$ ("Energy equals mass times the square of the speed of light (c)").

Some Typography   St. John vs. St. John. Fig. 5 vs. Fig. 5. Vice-president. Monday–Tuesday. Wait — I have an idea. "Wrong Quote" vs. "Correct Quote".

3 Some Mathematics

Definition 1 (Surjectivity)   Let $f : A \to B$ be a function from A to B. Then f is *surjective* if for every b in B there is some a in A such that f applied to a yields b, formally:

$$\forall b \in B. \; \exists a \in A. \; f(a) = b.$$

$\square$

Theorem 1 (Composition of Surjective Functions) *Let $f: A \to B$ and $g: B \to C$ be surjective functions (see Definition 1). Then the function $f \circ g: A \to C$ (defined as $(f \circ g)(a) := g(f(a))$) is surjective.* □

PROOF Left as an exercise to the reader. ■

4 Some Programs

Algorithm 1 Compute the set P of all primes less than equal $n \in \mathbb{N}$

Require: $n \in \mathbb{N}$
Ensure: $P = \{p \mid p \in \mathbb{N} \land p \le n \land isPrime(p)\}$
  $P \leftarrow \emptyset$
  $C \leftarrow \{2, \dots, n\}$
  while $C \ne \emptyset$ **do**
    $p \leftarrow \min(C)$
    $P \leftarrow P \cup \{p\}$
    $C \leftarrow \{c \in C : p \nmid c\}$
  end while

See Algorithm 1.

```cpp
// HelloWorld.cpp
#include <iostream>
using namespace std;

int main() {
  char message[] = "Hello, World!";
  for (int i=0; i<10; i++)
    cout << message << "\n";
}
```

```cpp
// HelloWorld.cpp
#include <iostream>
using namespace std;

int main() {
  char message[] = "Hello, World!";
  for (int i=0; i<10; i++)
    cout << message << "\n";
}
```

5 Conclusions

Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla. Bla. Bla, bla, bla bla.

References

[1] Wolfgang Ahrendt, Bernhard Beckert, Richard Bubel, Reiner Hähnle, Peter H. Schmitt, and Mattias Ulbrich, editors. *Deductive Software Verification —- The KeY Book: From Theory to Practice*, volume 10001 of *Lecture Notes in Computer Science*. Springer, Berlin, 2018.

[2] Mike Barnett, K. Rustan M. Leino, and Wolfram Schulte. The Spec# Programming System: An Overview. In Mihaela Bobaru, Klaus Havelund, Gerard J. Holzmann, and Rajeev Joshi, editors, *Construction and Analysis of Safe, Secure, and Interoperable Smart Devices (CASSIS 2004), Marseille, France, March 10-14, 2004*, volume 3362 of *Lecture Notes in Computer Science*, pages 49–69. Springer, Berlin, Germany, 2004.

[3] David R. Cok. OpenJML: JML for Java 7 by Extending OpenJDK. In Mihaela Bobaru, Klaus Havelund, Gerard J. Holzmann, and Rajeev Joshi, editors, *NASA Formal Methods (NFM 2011), Pasadena, CA, USA, April 18–20, 2011*, volume 6617 of *Lecture Notes in Computer Science*, pages 472–479. Springer, Berlin, Germany, 2011.

[4] K. Rustan M. Leino. Dafny: An Automatic Program Verifier for Functional Correctness. In Edmund M. Clarke and Andrei Voronkov, editors, *Logic Programming and Automated Reasoning (LPAR-16), Dakar, Senegal, April 25–May 1, 2010*, volume 6355 of *Lecture Notes in Computer Science*, pages 348–370. Springer, Berlin, Germany, 2010.

[5] Wolfgang Schreiner. Validating Mathematical Theories and Algorithms with RISCAL. In F. Rabe, W. Farmer, G. Passmore, and A. Youssef, editors, *CICM 2018, 11th Conference on Intelligent Computer Mathematics, Hagenberg, Austria, August 13–17*, volume 11006 of *Lecture Notes in Computer Science/Lecture Notes in Artificial Intelligence*, pages 248–254. Springer, Berlin, 2018.

[6] Wolfgang Schreiner, Alexander Brunhuemer, and Christoph Fürst. Teaching the Formalization of Mathematical Theories and Algorithms via the Automatic Checking of Finite Models. In Pedro Quaresma and Walther Neuper, editors, *Post-Proceedings ThEdu'17, Theorem proving components for Educational software, Gothenburg, Sweden, August 6, 2017*, volume 267 of *EPTCS*, pages 120–139, 2018.