## INSTITUTSSEMINAR

Am Donnerstag, dem 24. August 2023, spricht um 11:00 Uhr im Raum Z 2073

## Prof. Martin Dietzfelbinger

zum Thema:

## "On Hashing by (Random) Equations"

## Zusammenfassung:

The talk will consider aspects of the following setup:
Assume for each (key) $x$ from a set $U$ (the universe) a vector $a_{-} x=\left(a_{-}\{x, 0\}, \ldots, a_{-}\{x,\{m-1\}\}\right)$ has been chosen. Given a list $Z=\left(z_{-}\right) \_\{0<=i<m\}$ of vectors in $\backslash\{\overline{0}, 1 \backslash\} \wedge^{-} r$ we obtain a mapping phi_Z: U -> to $\{0,1\}^{\wedge} r$ r, phi $(x)=<a_{-} x, Z>:=X O R \_\{0<=i<m\} a_{-}\{x, i\} . z \_$
(XOR denotes bitwise exclusive-or, and the dot denotes multiplication.)
The simplest way for creating a data structure for calculating phi_Z is to store $Z$ in an array $Z[0 . . \mathrm{m}-1]$ and answer a query for $x$ by returning $\left\langle a_{-} x, Z\right\rangle$. The length $m$ of the array should be ( $1+$ \varepsilon) $n$ for some small \varepsilon, and calculating this inner product should be fast. In the focus of the talk is the case where for all or for most of the sets S\subseteq \univ of a certain size $n$ the vectors $a_{-} x$, $x$ in $S$, are linearly independent. Choosing $Z$ at random will lead to hash families of various degrees of independence. We will be mostly interested in the case where a x , $x$ in $U$, are chosen independently at random from $\{0,1\}^{\wedge} m$, according to some distribution $D$. We wish to construct (static) retrieval data structures, which means that a subset $S$ of $U$ and some mapping f: $S->\{0,1\} \wedge r$ are given, and the task is to find $Z$ such that the restriction of phi_Z to $S$ is $f$. For creating such a data structure it is necessary to solve the linear system $\left(a_{-} \mathrm{x}\right)_{\_}\{\mathrm{x}$ in S$\} . \mathrm{Z}=(\mathrm{f}(\mathrm{x}))_{-}\{\mathrm{x}$ in S$\}$ for Z . Two problems are central: Under what conditions on m and $D$ can we expect the vectors $a_{-} x, x$ in $S$, to be linearly independent, and how can we arrange things so that in this case the system can be solved fast, in particular in time close to linear (in n , assuming a reasonable machine model)? Solutions to these problems, some classical and others that have emerged only in recent years, will be discussed.
(Slides in English, talk in English or German.)
Alle Interessenten sind herzlich eingeladen.

