



CALL FOR PAPERS

ICFPG 2022
Apr 08-09, 2022
Athens, Greece

The International Research Conference is a federated organization dedicated to bringing together a significant number of diverse scholarly events for presentation within the conference program. Events will run over a span of time during the conference depending on the number and length of the presentations.

ICFPG 2022 : International Conference on Functional Programming and Games is the premier interdisciplinary forum for the presentation of new advances and research results in the fields of Functional Programming and Games. The conference will bring together leading academic scientists, researchers and scholars in the domain of interest from around the world. Topics of interest for submission include, but are not limited to:

Functional programming and multicore/manycore computing
Functional programming in the cloud
Functional programming in education
High performance functional computing
Extra-functional (behavioural) properties of functional programs
Dependently typed functional programming
Validation and verification of functional programs
Using functional techniques to verify/reason about imperative/object-oriented programs
Debugging for functional languages
Functional programming in different application areas: security, mobility, telecommunications applications, embedded systems, global computing, grids, etc.
Interoperability with imperative programming languages
Novel memory management techniques

Program transformation techniques
Empirical performance studies
Abstract/virtual machines and compilers for functional languages
New implementation strategies
Any new emerging trend in the functional programming area
Language Design: concurrency and distribution; modules; components and composition; metaprogramming; interoperability; type systems; relations to imperative, object-oriented, or logic programming
Implementation: abstract machines; virtual machines; interpretation; compilation; compile-time and run-time optimization; memory management; multi-threading; exploiting parallel hardware; interfaces to foreign functions, services, components, or low-level machine resources
Software-Development Techniques: algorithms and data structures; design patterns; specification; verification; validation; proof assistants; debugging; testing; tracing; profiling
Foundations: formal semantics; lambda calculus; rewriting; type theory; monads; continuations; control; state; effects; program verification; dependent types
Analysis and Transformation: control-flow; data-flow; abstract interpretation; partial evaluation; program calculation
Applications and Domain-Specific Languages: symbolic computing; formal-methods tools; artificial intelligence; systems programming; distributed-systems and web programming; hardware design; databases; XML processing; scientific and numerical computing; graphical user interfaces; multimedia programming; scripting; system administration; security

