On-line auctions implementation

The contest proposed by @Futurizt

There is a clear need for efficient implementation of different Auction types in Free TON. Since many use cases require auction mechanics, we need a reference implementation of many auction types in-line with distributed smart contract architecture. Here it is proposed the Contest to fill that gap.

Contest participants will compete to create, test and deploy smart contracts implementing different types of auctions.

Contest period

Contest Dates: 31 March 2021 — 30 April 2021
Judging: 01 May 2021 — 16 May 2021

What types of auctions?

Since we have already implemented Vickry auctions in DeNS contest we should concentrate on the following auction types and their variants:

- English auction
- Dutch

Both auction types should be implemented for forward, reversed auction types and the blind variant (which technically are considered a separate type but we will use the English and Dutch names for simplicity) should be submitted with at least a commit-reveal scheme or zero-knowledge proofs.

What are architectural guidelines?

All submitted auction implementations should be composable, meaning the combined auction systems supporting governance contracts, etc, could be set up using the submissions. For example a cascading auctions, auctions with reserved or non reserved price and so on could be set up with minimal additions.

Auctions must be scalable and should not rely on a single contract. The Auction Root contract should be able to deploy an infinite amount of atomic auctions. Bids should be submitted as a separate contract each.

Interfaces

Contracts must interface with TON Crystal’s native cryptocurrency, the Transfer function in DePools as well as TIP-3 tokens.

Auctions must emit on-chain events reporting important state changes that could be used by other smart contracts.

What programming languages could be used?

Solidity or C++ or combination of these must be used

What should submission include?

- Source code (open source, Free Software licence).
- Test scripts for all functionality using one of the existing frameworks (should cover all functions, run and pass):
    - TS-4 in Python from TonLabs;
    - Echpochmak fron Wintexpro;
    - TON testing suite or Locklift from Broxus.
- Description using any of the following languages: TLA+ family, TL-B, UML, Prolog family.
- DeBots following the latest specifications to all user interactions with auction smart contracts.
- All smart contracts and DeBots should be deployed to DevNet, addresses provided in the submission, operational.

SetCode Policy

Contracts may have the SetCode function but the clean no SetCode variant should be provided as well.

Defense of contest works

At the end of the applications acceptance, AMA-session will be appointed for participants, jurors and everyone else. At this session, each contestant has to present his/her work.

A video presentation of the work is highly desirable. There are no strict requirements for the presentation. It can be prepared in any form (slide show, program text demonstration, video performance, etc.).

The presentation language is English or Russian. The presentation time should not exceed 10-20 minutes.

If a contestant cannot present the work on-line, he/she can make a video recording and publish it on YouTub.

Winners criteria

Best implementation of most contest types according to provided architectural guidelines and contest terms.

Proposed prizes

- 1st place — 50 000 TON
- 2nd place — 25 000 TON
- 3rd place — 10 000 TON
- 4th place — 2 500 TON
- 5th place — 2 500 TON

Voting

Only qualified jurors, able to understand and evaluate smart contracts on Free TON should judge.
Each juror will vote by rating each submission on a scale of 1 to 10 or can choose to reject it if it does not meet requirements, or they can choose to abstain from voting if they feel unqualified to judge.
Jurors will provide feedback on your submissions.
Duplicate, sub par, incomplete or inappropriate submissions will be rejected.

The jury

Telegram ID	Public key
@lailune	fb2fe560bfbdda910798e1365d9419ff6e0a75ed5262410b714f162434a88af6
@d_borisenkov	09e78ab9ceb5a0df34ecd4d70822848eba2d8cfc63fa05ef613b5d9a373b2a99
@inyellowbus	f0c02482ac2a54691aba1e2c38b6cd7fd3708f9eb015058c7d85a939646715e7
@Custler	2c0ec55a109eb466d9db5ee7c3adb075e77627ade83ae17cea847671ab8f0a85
@RRoman_Tver	49c1c8453240fced11141a690870924fb4d9e9eeb8ea6fae54b8549677fb8bd5
@anovi	6bf867e7773038f9491b5d397cb7da8a20e112b45db9523cd8dae9deb608fb5a
@ded_mrz	9e808b1540babb85c428b9197b8df87860882d2db70607dfa134774a0513db30
@aleksandr_h	3bdb99452a6ed9138af385d035e0967250d0e2da6f58b90245f285daf250915c
@zxcat	0f07a7cb924c7420520d0d98afad87d9b5e1765920fda698c22da6d0cd3354b9

@bivanovsky	1a99622e54b4e87d603dd87c9cc936b388b2a0e1979bb56d4039cfad0fbadc8c
@get_username	d89e20f2e164f4e64a3637cc6926baf66482bf8c875e4e7eebedd6f28024998c
@pafaul	7657b6b62ec6846e6341315ba3b8d611afe2ef79e8ee3214d45cc4ede3c50972

Jury rewards

An amount equal to 5 % of all total tokens actually awarded and distributed will go to each juror for performing their civic duty to the community and taking the time to judge each submission and provide feedback.