Funcons-beta: Abrupting *

The PLanCompS Project

`Abrupting.cbs` | PLAIN | PRETTY

Abruptly terminating

[*Funcon* stuck
Entity abrupted
Funcon finalise-abrupting
Funcon abrupt
Funcon handle-abrupt
Funcon finally]

Meta-variables T, T', T'' <: values

Funcon stuck : ⇒ empty-type

stuck does not have any computation. It is used to represent the result of a transition that causes the computation to terminate abruptly.

Entity $\_ \xrightarrow{\text{abrupted}(\_:\text{values}^?)} \_$

abrupted(V) in a label on a tranistion indicates abrupt termination for reason V. abrupted( ) indicates the absence of abrupt termination.

Funcon finalise-abrupting(X : ⇒ T) : ⇒ T | null-type
⇝ handle-abrupt(X, null-value)

finalise-abrupting(X) handles abrupt termination of X for any reason.

Funcon abrupt(_ : values) : ⇒ empty-type

abrupt(V) terminates abruptly for reason V.

Rule abrupt(V : values) $\xrightarrow{\text{abrupted}(V)}$ stuck

Funcon handle-abrupt(_ : T' ⇒ T, _ : T'' ⇒ T) : T' ⇒ T

---

*Suggestions for improvement: `plancomps@gmail.com`.
Reports of issues: `https://github.com/plancomps/CBS-beta/issues`.

handle-abrupt(X, Y) first evaluates X. If X terminates normally with value V, then V is returned and Y is ignored. If X terminates abruptly for reason V, then Y is executed with V as given value.

handle-abrupt(X, Y) is associative, with abrupt(given) as left and right unit. handle-abrupt$(X,$ else$(Y,$ abrupt(given))) ensures propagation of abrupt termination for the given reason if Y fails

Rule $$\frac{X \xrightarrow{\text{abrupted}(\ )} X'}{\text{handle-abrupt}(X, Y) \xrightarrow{\text{abrupted}(\ )} \text{handle-abrupt}(X', Y)}$$

Rule $$\frac{X \xrightarrow{\text{abrupted}(V:T'')} X'}{\text{handle-abrupt}(X, Y) \xrightarrow{\text{abrupted}(\ )} \text{give}(V, Y)}$$

Rule $\text{handle-abrupt}(V : T, Y) \rightsquigarrow V$

Funcon $\text{finally}(\_ :\Rightarrow T, \_ :\Rightarrow \text{null-type}) :\Rightarrow T$

finally(X, Y) first executes X. If X terminates normally with value V, then Y is executed before terminating normally with value V. If X terminates abruptly for reason V, then Y is executed before terminating abruptly with the same reason V.

Rule $$\frac{X \xrightarrow{\text{abrupted}(\ )} X'}{\text{finally}(X, Y) \xrightarrow{\text{abrupted}(\ )} \text{finally}(X', Y)}$$

Rule $$\frac{X \xrightarrow{\text{abrupted}(V:\text{values})} X'}{\text{finally}(X, Y) \xrightarrow{\text{abrupted}(\ )} \text{sequential}(Y, \text{abrupt}(V))}$$

Rule $\text{finally}(V : T, Y) \rightsquigarrow \text{sequential}(Y, V)$