$$\text{————————— MODULE } pluscal\_exp \text{ —————————}$$

EXTENDS *Naturals, Integers, TLC*

CONSTANT *MAXINT*, $x10$, $x20$, *MININT*

$typeInt(u) \triangleq u \in Int$

$pre \triangleq x10 \in Nat \land x20 \in Nat \land x10 \neq 0$

precondition
ASSUME *pre*

```
– algorithm Power {
  variables
        x1 = x10;
        x2 = x20;
        y1;
        y2;
        y3;
        z;
{
   l0 : assert x1 = x10 ∧ x2 = x20 ∧ typeInt(x1) ∧ typeInt(x2) ∧ typeInt(y1) ∧ typeInt(y2) ∧
   typeInt(y3) ∧ typeInbt(z);
   print⟨x1, x2⟩;
   y1 := x1;
   y2 := x2;
   y3 := 1;
   l1 : assert x1 = x10 ∧ x2 = x20 ∧ y2 ≥ 0 ∧ y3 * y1^y2 = x1^x2 ∧ typeInt(x1) ∧ typeInt(x2) ∧
   typeInt(y1) ∧ typeInt(y2) ∧ typeInt(y3) ∧ typeInt(z);

while(y2 ≠ 0){

     l2 : assert y2 \neq 0 ∧ y3 * y1^y2 = x1^x2 ∧ ∧ typeInt(x1) ∧ typeInt(x2) ∧ typeInt(y1) ∧
     typeInt(y2) ∧ typeInt(y3) ∧ typeInt(z);


if(y2%2 ≠ 0){
        l3 : y2 := y2 − 1;
        l4 : y3 := y3 * y1;
        l5 : skip;
     };
     l6 : y1 := y1 * y1;       l7 : y2 := y2 ÷ 2;
     l8 : skip;
   };
   l9 : z := y3;
   l10 : print⟨x1, x2, z⟩;
}

}
```

1

$L \triangleq \{$"l0", "l1"$\}$

$D \triangleq MININT \mathinner{\ldotp\ldotp} MAXINT$

$DD(X) \triangleq X = defaultInitValue \Rightarrow X \in D$

$i \triangleq$
  $\land pc \in L$
  $\land DD(y1) \land DD(y2) \land DD(y3) \land DD(z)$
  $\land pc \quad =$ "l1" $\Rightarrow x1 = x10 \land x2 = x20 \land \;\; y2 \geq 0 \;\; \land y3 * y1^{y2} = x1^{x2} \land \land typeInt(x1) \land typeInt(x2) \land typeIn$

$Q1 \triangleq pc \neq$ "Done"

$Qpc \triangleq pc =$ "Done" $\Rightarrow z = x1^{x2}$