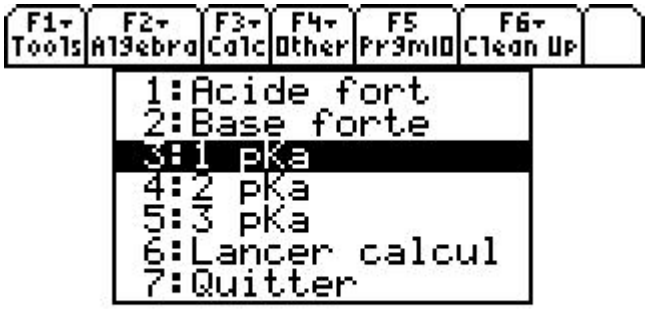
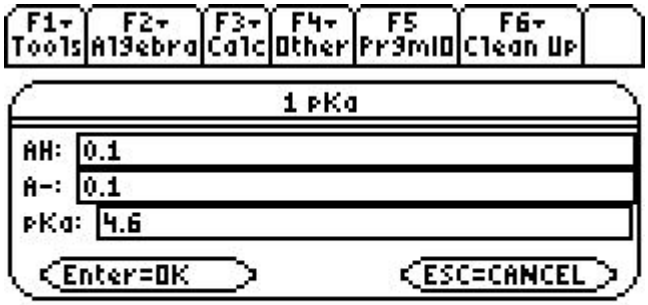
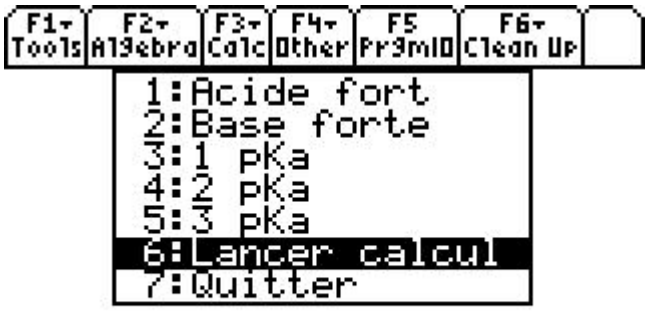
Ti89

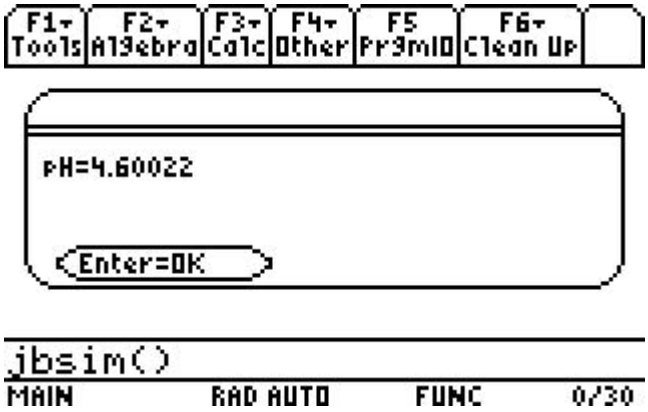
jbsim89

Un exemple pour un mélange tampon :










```
« "JBSIM 1.3 BY J.B." { "Calcul d'un pH" "Dosage" } 1.
  IF CHOOSE
  THEN { "Calcul d'un pH" "Dosage" } SWAP POS { Vd VM NVAC V H CALCpH EQPH }
PURGE { AcideFort
    « "Acide Fort" { { "C" "Concentration en acide fort" 0. 6. } } 1. { } {
}
      IF INFORM
      THEN OBJ→ DROP
      END -1. 0.
    » BaseForte
    « "Base Forte" { { "C" "Concentration en base forte" 0. 6. } } 1. { } {
}
      IF INFORM
      THEN OBJ→ DROP
      END 1. 0.
    » Acidite1
    « "1 ACIDITE" { { "CA" "Concentration en [AH]" 0. 6. } { "CB"
"Concentration en [A-]" 0. 6. } { "pKa" "-LOG(Constante d'acidite)" 0. 6. }
} 1. { } { }
      IF INFORM
      THEN OBJ→ DROP
      END → CA CB pKa
      « H CB * CA pKa NEG ALOG * - H pKa NEG ALOG + / 1.
      » 0.
    » Acidites2
    « "2 ACIDITES" { { "C1" "Concentration en AH2" 0. 6. } { "C2"
"Concentration en AH-" 0. 6. } { "C3" "Concentration en A2-" 0. 6. } {
"pKa1" "-LOG(Constante d'acidite 1)" 0. 6. } { "pKa2" "-LOG(Constante
d'acidite 2)" 0. 6. } } 2. { } { }
      IF INFORM
      THEN OBJ→ DROP
      END → C1 C2 C3 pKa1 pKa2
      « C2 2. C1 * + H C3 * H C1 * - pKa2 ALOG * - 2. H 2. ^ * C3 * H 2. ^
C2 * + pKa2 pKa1 + ALOG * - NEG 1. H pKa2 ALOG * + H 2. ^ pKa2 pKa1 + ALOG *
+ / 1.
      » 0.
    » Acidites3
    « "3 ACIDITES" { { "C1" "Concentration en H3A" 0. 6. } { "C2"
"Concentration en H2A-" 0. 6. } { "C3" "Concentration en HA2-" 0. 6. } {
```

```
"C4" "Concentration en A3-" 0. 6. } { "pKa1" "-LOG(Constante d'acidite 1)"
0. 6. } { "pKa2" "-LOG(Constante d'acidite 2)" 0. 6. } { "pKa3" "-
LOG(Constante d'acidite 3)" 0. 6. } } 2. { } { }
      IF INFORM
      THEN OBJ→ DROP
      END → C1 C2 C3 C4 pKa1 pKa2 pKa3
      « -1. 10. pKa1 NEG pKa2 - pKa3 - ^ 10. pKa2 NEG pKa1 - ^ H * + 10.
pKa1 NEG ^ H 2. ^ * + H 3. ^ + / 3. 10. pKa2 NEG pKa3 - ^ * 2. 10. pKa2 NEG
^ * H * + H 2. ^ + * C1 C2 + C3 + C4 + * 10. pKa1 NEG ^ * C2 + 2. C3 * + 3.
C4 * + 1.
      » 0.
    » CH1
    « { AcideFort BaseForte Acidite1 Acidites2 Acidites3 Continuer } 1.
    » Calcul1pH
    « { Vd 10. } | { V 0. } | 'H' { 0. .0000001 1.E-14 } ROOT LOG NEG "pH"
→TAG 1.
    » Courbe
    « (0.,-2.) VM SORT DUP SIZE GET 14. R→C 'V' # 2h { (0.,0.) { 1. 1. } } {
FUNCTION } OBJ→ DROP 'Y' 7. →LIST 'PPAR' STO
      « 'EQPH' 'H' 0. H 1.E-14 3. →LIST ROOT LOG NEG
      » 'EQ' STO PICT PURGE DRAX DRAW
    » } → Ch LISTE
    « 'H-1.E-14/H' {
      «
        DO LISTE DUP CH1 POS 1. + GET "BECHER" SWAP EVAL
          IF CHOOSE
          THEN
            IF DUP Continuer SAME
            THEN DROP LISTE DUP Calcul1pH POS 1. + GET EVAL
            ELSE LISTE DUP 3. ROLL POS 1. + GET EVAL 3. ROLL 3. ROLL * 3.
ROLL + SWAP
            END
          ELSE DROP 1.
          END
        UNTIL
        END
      »
      «
        DO LISTE DUP CH1 POS 1. + GET "BECHER" SWAP EVAL
          IF CHOOSE
          THEN
            IF DUP Continuer SAME
            THEN DROP
              DO LISTE DUP CH1 POS 1. + GET "BURETTE" SWAP EVAL
                IF CHOOSE
                THEN
                  IF DUP Continuer SAME NOT
                  THEN LISTE DUP 3. ROLL POS 1. + GET EVAL 3. ROLL V * Vd V
+ / 3. ROLL * 3. ROLL + SWAP
                  ELSE DROP "VOLUME" { { "LIST V" "LISTE DES PH A CALCULER"
5. } { "Vd" "VOLUME DE SOLUTION A DOSER" 0. 6. } } 1. { } { }
```

```
                    IF INFORM
                    THEN OBJ→ DROP 'Vd' STO DUP 'VM' STO SIZE 'NVAC' STO
EVAL 'EQPH' STO 1. 1.
                    ELSE DROP 1.
                    END
                  END
                ELSE DROP 1.
                END
              UNTIL
              END
            ELSE LISTE DUP 3. ROLL POS 1. + GET EVAL 3. ROLL Vd * Vd V + /
3. ROLL * 3. ROLL + SWAP
            END
          ELSE DROP 1.
          END
        UNTIL
        END 0. 'H' STO
        « 'EQPH' 'H' 0. H 1.E-14 3. →LIST ROOT LOG NEG
        » 'CALCpH' STO "   +--------------+" 3. DISP "   | Calcul de pH |"
4. DISP "   | en cours ... |" 5. DISP "   +--------------+" 6. DISP 1. NVAC
        FOR M VM M GET DUP 'V' STO CALCpH
        NEXT NVAC 2. 2. →LIST →ARRY "Courbe ?" { "Oui" "Non" } 1.
        IF CHOOSE
        THEN
          IF "Oui" SAME
          THEN LISTE DUP Courbe POS 1. + GET EVAL
          END
        END { Vd VM NVAC } PURGE
      » } Ch GET EVAL
    »
  END
»
```