**2.1.3**

**Chaine** valeurAConvertir

**Reel** distanceEnReel(100)

**Chaine** uniteConversion(100)

**Entier** nbValeurs

**Entier** indexValeurCourrante

**Booleen** veutQuitter

**Booleen** veutConvertir

veutQuitter <-- false

veutConvertir <-- false

**Ecrire** « Programme de conversion km<->mi »

**Répéter**

veutConvertir<-- false

veutQuitter <-- false

nbValeurs <-- 0

indexValeurCourrante <--1

**Répéter**

nbValeurs<-- nbValeurs+1

**Ecrire** « rentrez une valeur à convertir et son unité séparées par un espace »

**Lire** valeurAConvertir

**SI** (valeurAConvertir == «go » ) **ALORS**

veutConvertir <-- true

**SINON SI** (valeurAConvertir== « quit ») **ALORS**

veutQuitter <-- true

**SINON** SeparerValeurs(valeurAConvertir,distanceEnReel(nbValeurs),uniteConversion(nbValeurs))

**FINSI**

**Jusqu’à** (veutQuitter == false ET veutConvertir == false)

**SI** (veutQuitter == false) **ALORS**

**pour** indexValeurCourrante de 1 à nbValeurs **faire**

Convertir(distanceEnReel(indexValeurCourrante) ,uniteConversion(indexValeurCourrante))

**FIN POUR**

**FINSI**

**Jusqu’à**(veutQuitter == false)

**Void** **FONCTION** Convertir (**VAL Réel** distanceEnReel**, VAR chaine** uniteDistance**)**

**Entier** indexCourant

**Réel** valeurConvertie

**SI** (unitesDistance(indexCourant) == « mi ») **alors**

valeurConvertie<--distanceEnReel\*1.609

**Ecrire** distanceEnReel, « mi = », uniteDistance, « km . »

**SINON**

valeurConvertie<--distanceEnReel/1.609

**Ecrire** distanceEnReel, « km = », uniteDistance, « mi . »

**FINSI**

**FIN FONCTION**

**Void FONCTION** SeparerValeur**s(VAL chaine** distanceEnChaine**, VAR Réel** affectationReel**, VAR chaine** affectationChaine**)**

**Réel** distance

**Chaine** uniteMesure

Entien indexChaineDistance

indexChaineDistance <-- 1

distance <-- 0

**SI** (|chaine| > 3) **ALORS**

uniteMesure <-- distanceEnChaine(|distanceEnChaine|-1)

uniteMesure <-- uniteMesure + distanceEnChaine(|distanceEnChaine|)

**SI** (uniteMesure == « km ») **ALORS**

**TANTQUE** (indexChaineDistance <= |distanceEnChaine|-3)

distance <-- distance + distanceEnChaine(indexChaineDistance)

**FIN TANTQUE**

affectationReel <-- distance

affectationChaine <-- uniteMesure

**SINON SI** (uniteMesure == « mi ») **ALORS**

**TANTQUE** (indexChaineDistance <= |distanceEnChaine|-3)

distance <-- distance + distanceEnChaine(indexChaineDistance)

**FIN TANTQUE**

affectationReel <-- distance

affectationChaine <-- uniteMesure

**SINON**

affectationReel<-- distanceEnChaine

affectationChaine<--« km »

**FINSI**

**SINON**

affectationReel <-- distanceEnChaine

affectationChaine<--« km »

**FINSI**

**FIN FONCTION**