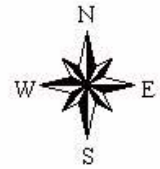
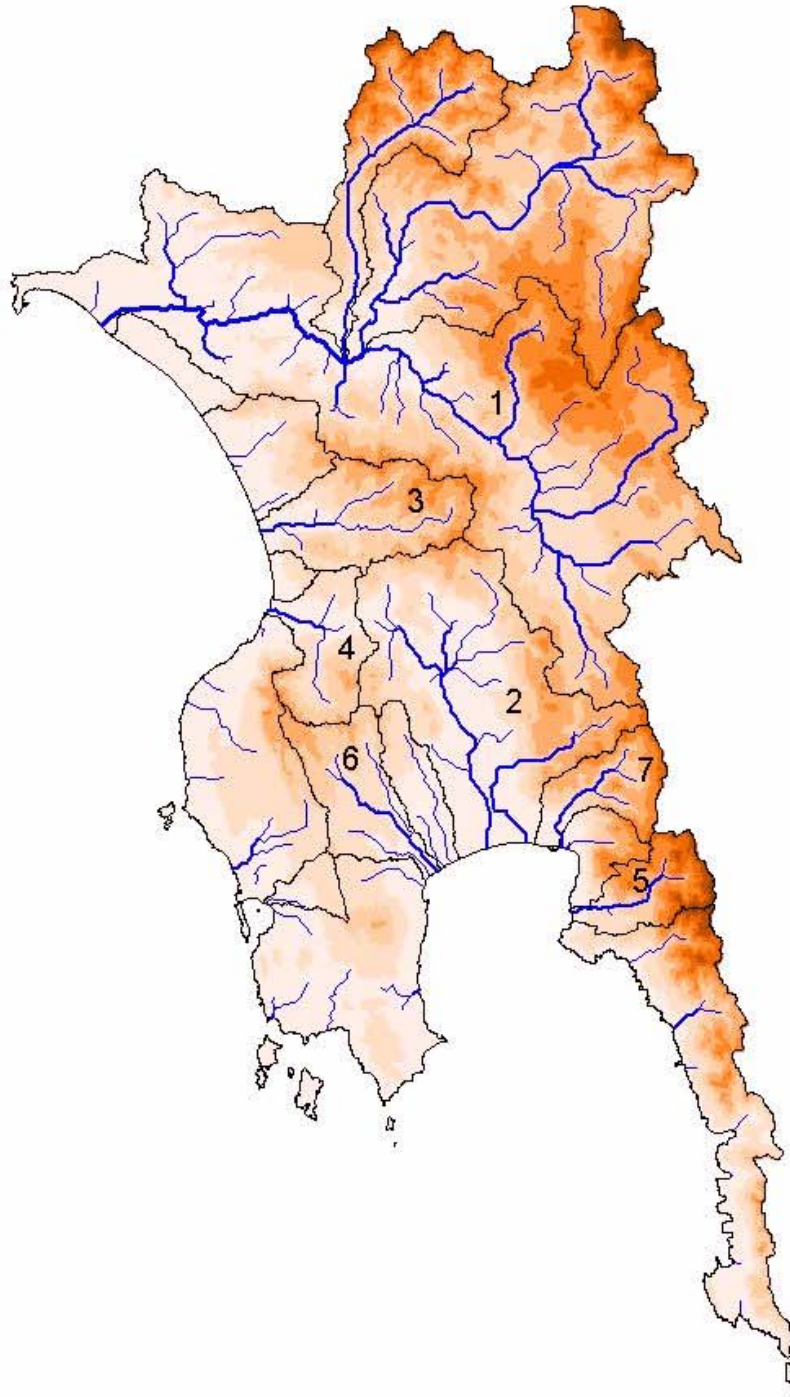


# **Παράρτημα Α**

## **Υδατικά διαμερίσματα**

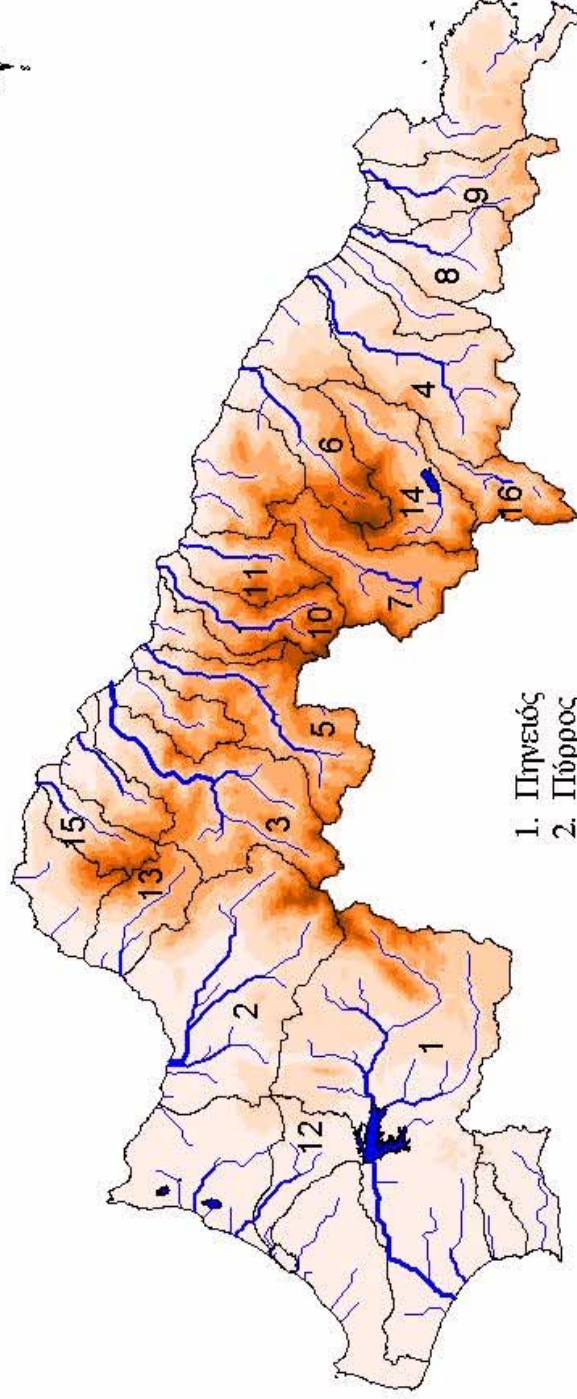
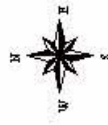
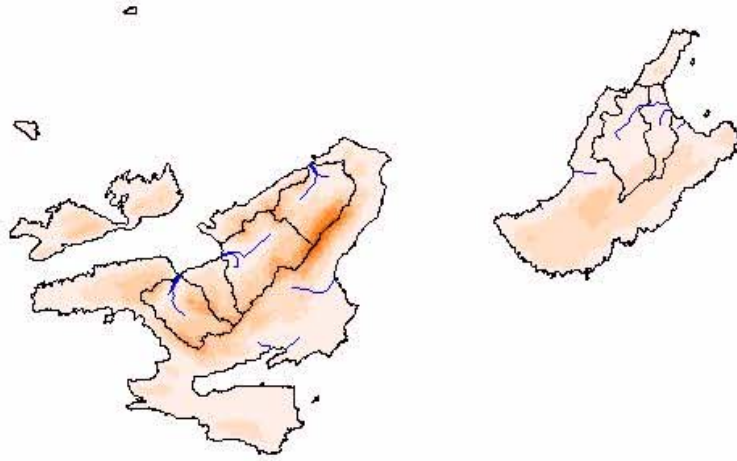
# Δυτική Πελοπόννησος



1. Αλφειός
2. Πάμισος
3. Νέδα
4. Καλό Νερό
5. Ρέμα Κάμπου
6. Βέλικας
7. Νέδωνας



# Βόρεια Πελοπόννησος

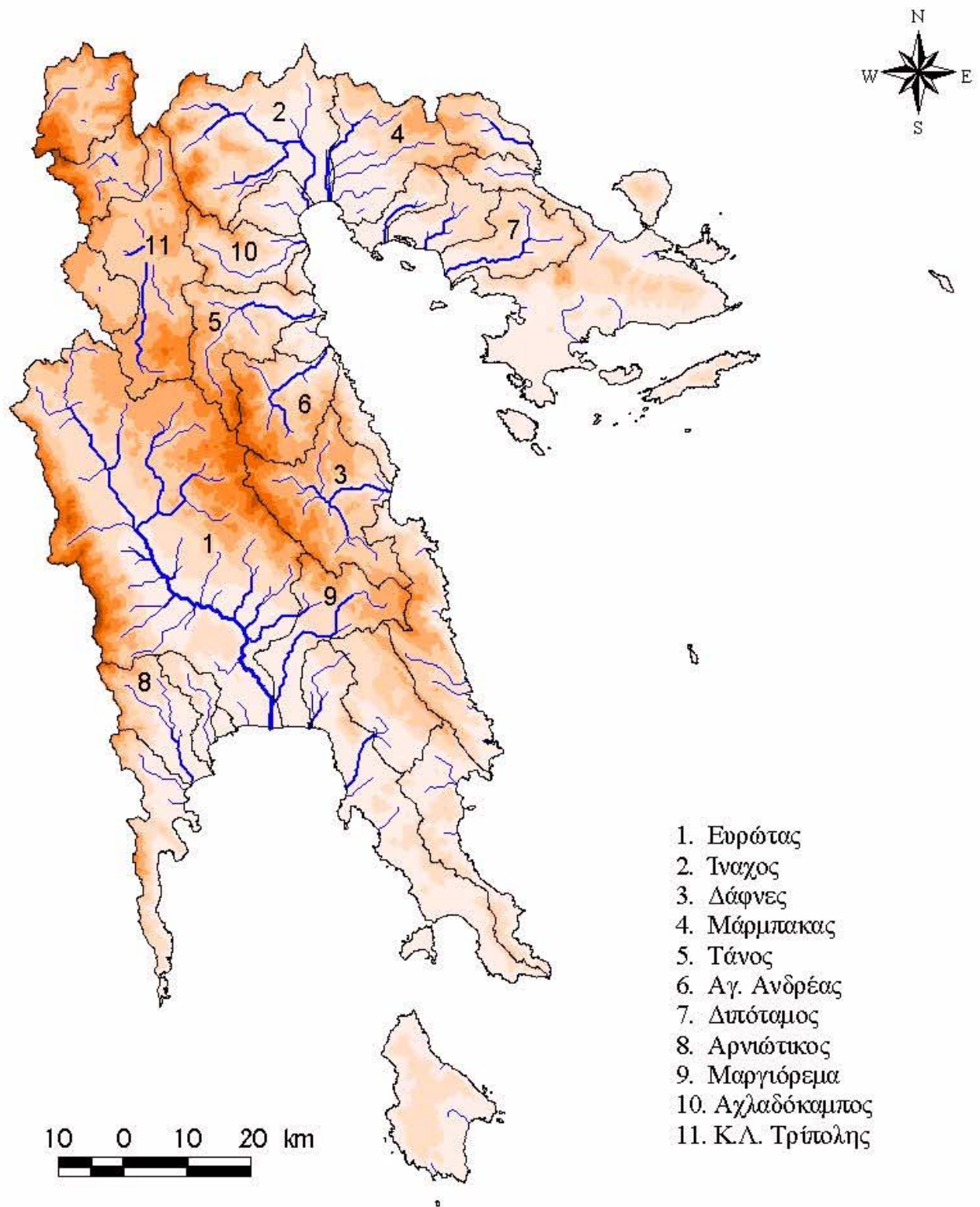


1. Πηνειός
2. Πύρρος
3. Σελινούντιας
4. Ασωπός
5. Βουραϊκός
6. Σύθας
7. Κ.Λ. Φενεού
8. Ραχιάνης
9. Κορίνθου
10. Κρέθης
11. Κριός
12. Βέργας
13. Γλαυκός
14. Κ.Λ. Στυμφαλίας(Αν)
15. Φοίνικας
16. Κ.Λ. Αλλάς

20 0 20 Kilometers

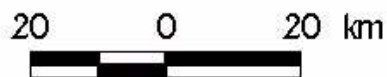
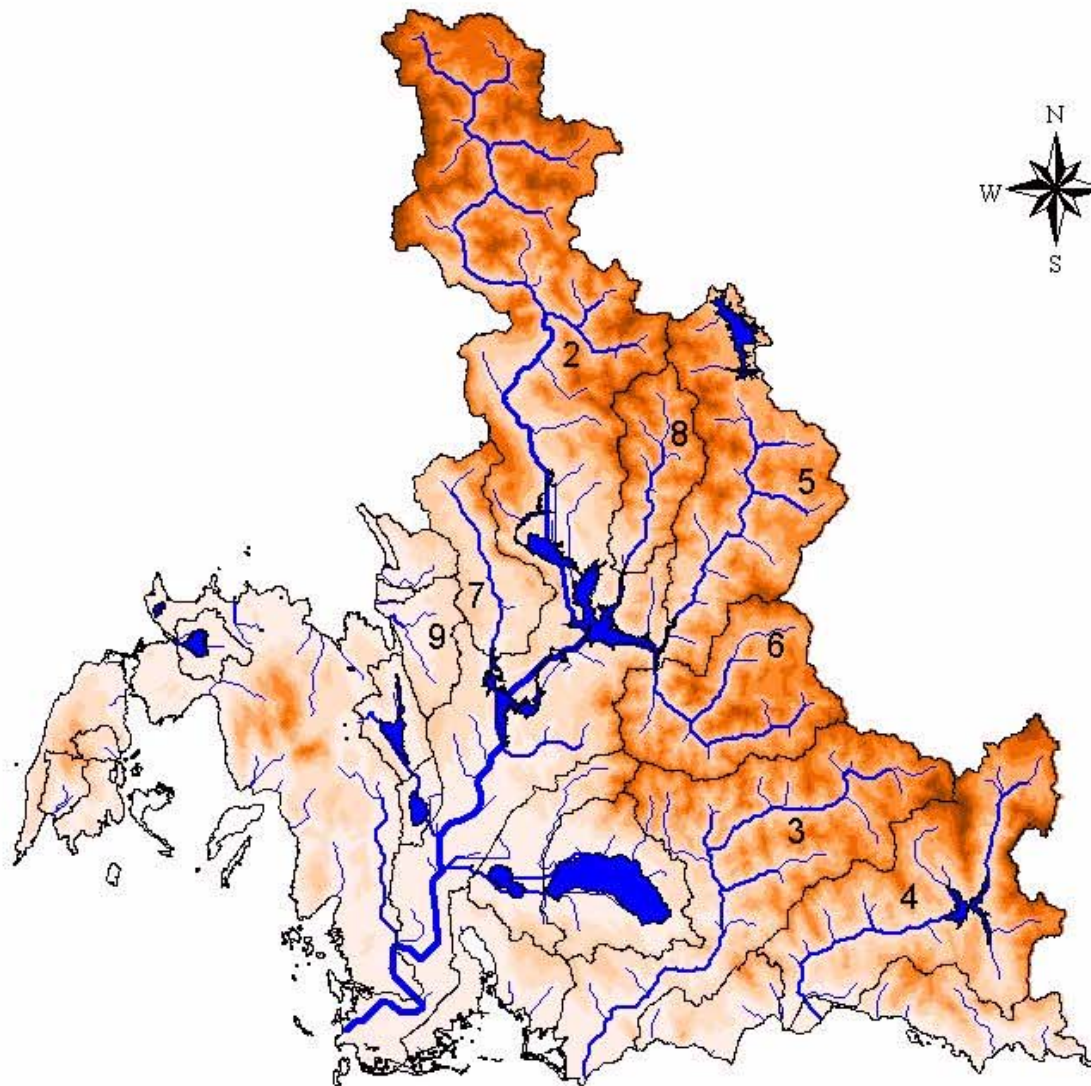


# Ανατολική Πελοπόννησος



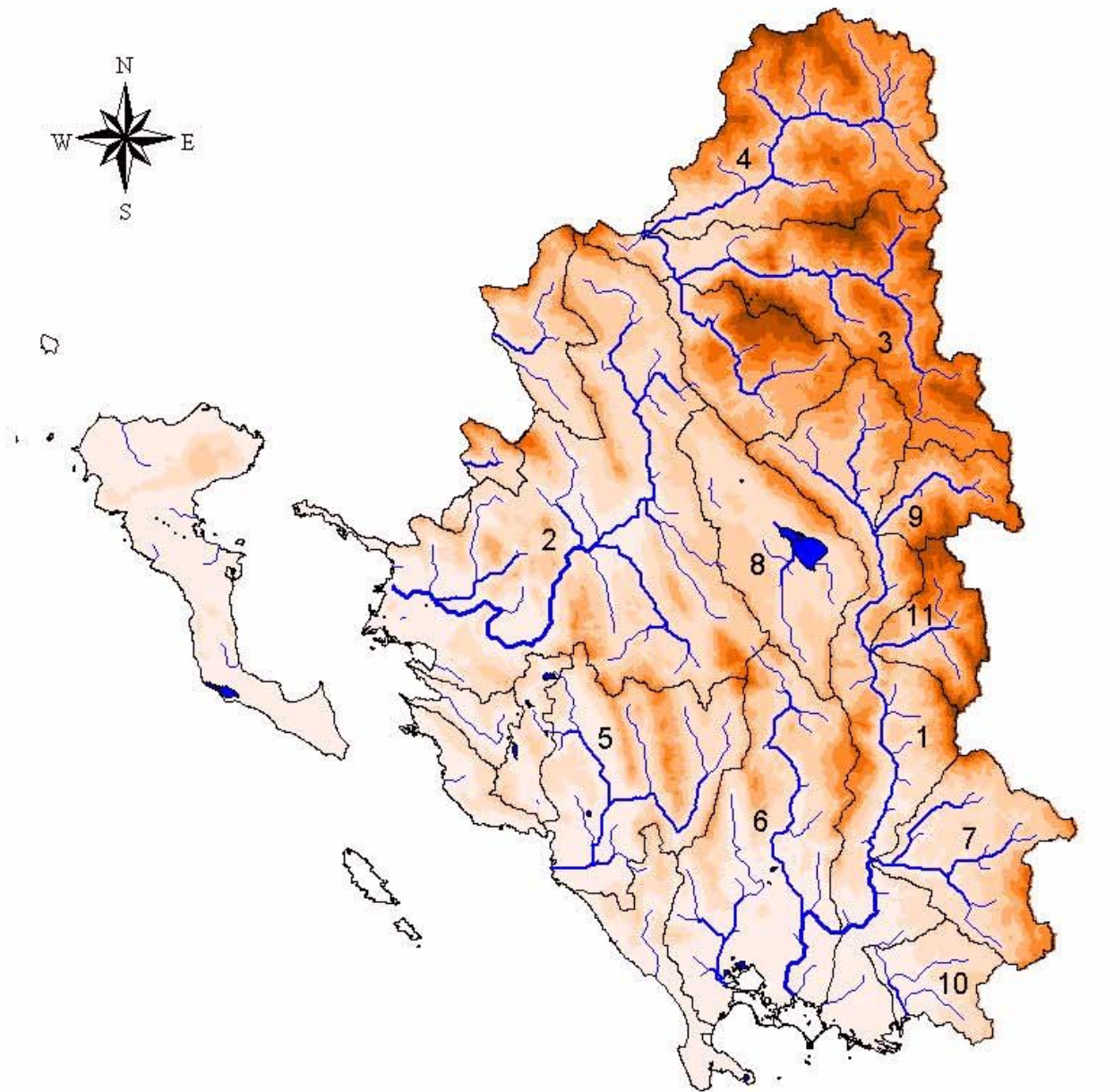
1. Ευρώτας
2. Ίναχος
3. Δάφνες
4. Μάρμαρακας
5. Τάνος
6. Αγ. Ανδρέας
7. Διπτόταμος
8. Αρνιώτικος
9. Μαργιόρεμα
10. Αγλαδόκαμπος
11. Κ.Λ. Τρίτολης

# Δυτική Στερεά



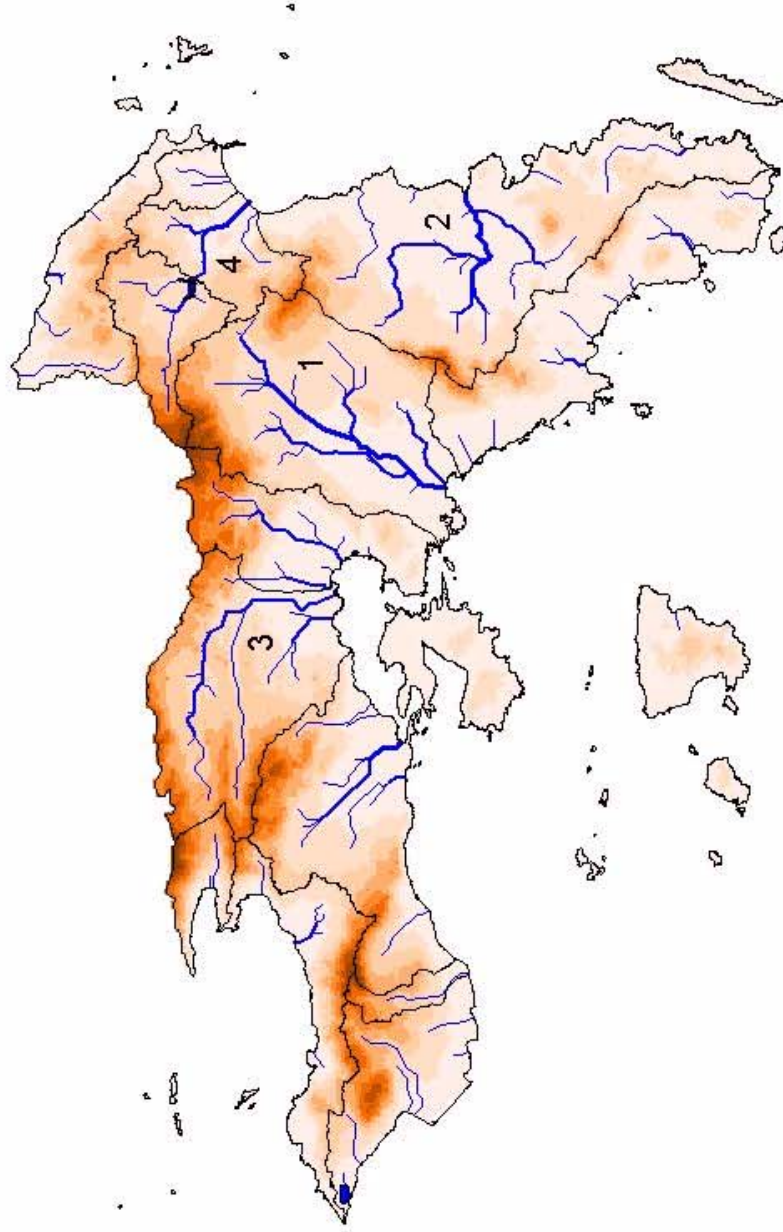
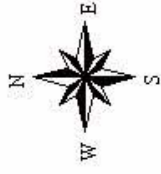
1. Αχελώος
2. Κρεμασιά
3. Εήνος
4. Μορνος
5. Μέγδοβας
6. Τρικεριώτης
7. Ιναχος
8. Αγραφιώτης
9. Ξηροπόταμος

# Ήπειρος



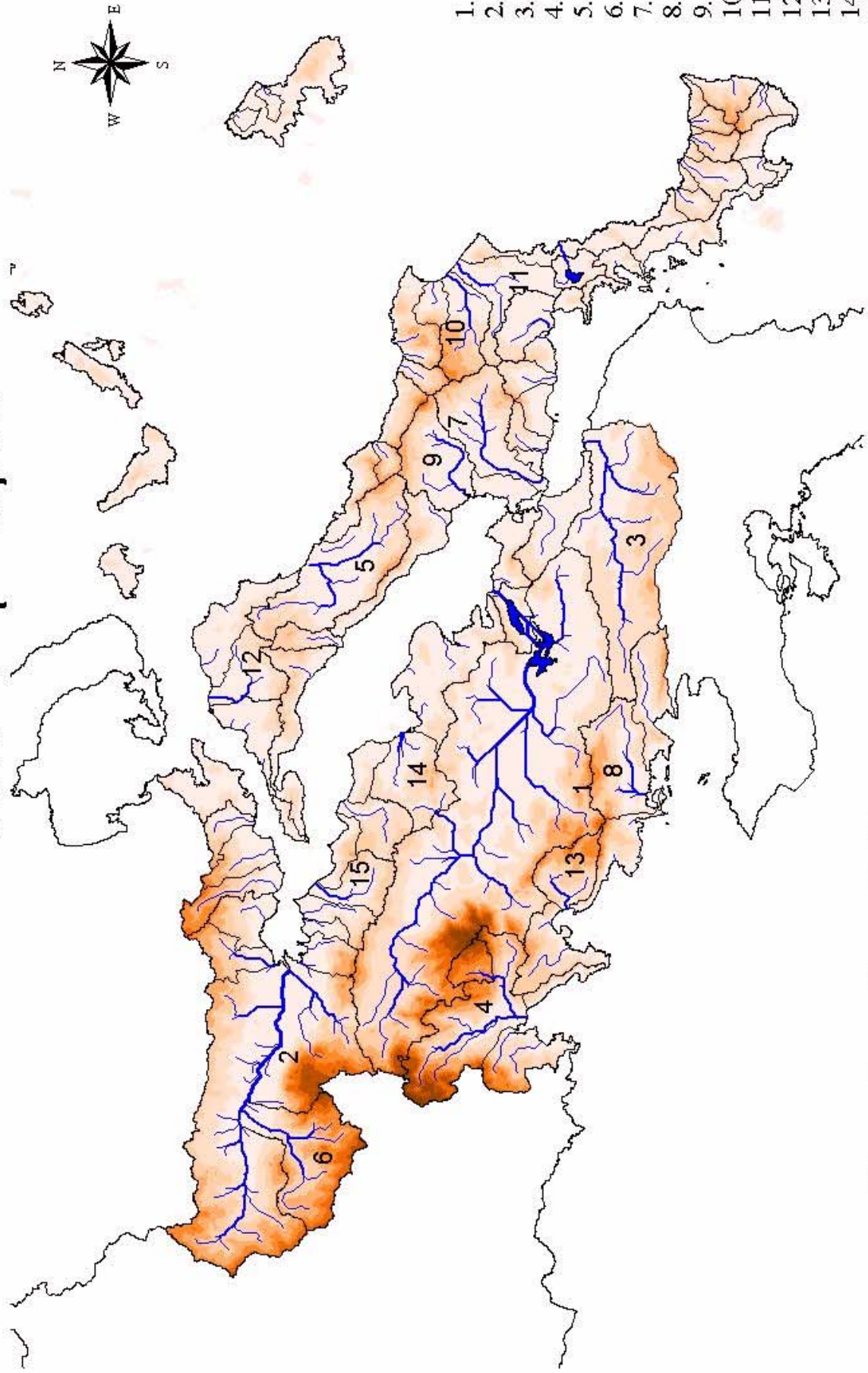
1. Αραχθος
2. Καλαμιάς
3. Αώος
4. Σαραντάπορος
5. Αχέρωντας
6. Λούρος
7. Καλεντίνη
8. Κ.Λ. Ιωαννίνων
9. Καλλαρίτικος
10. Κομποτάς
11. Μετσοβίτικος

# Αττική



1. Κηφισός
2. Υμηττός
3. Σερραυτελλότιμος
4. Χέραδρος

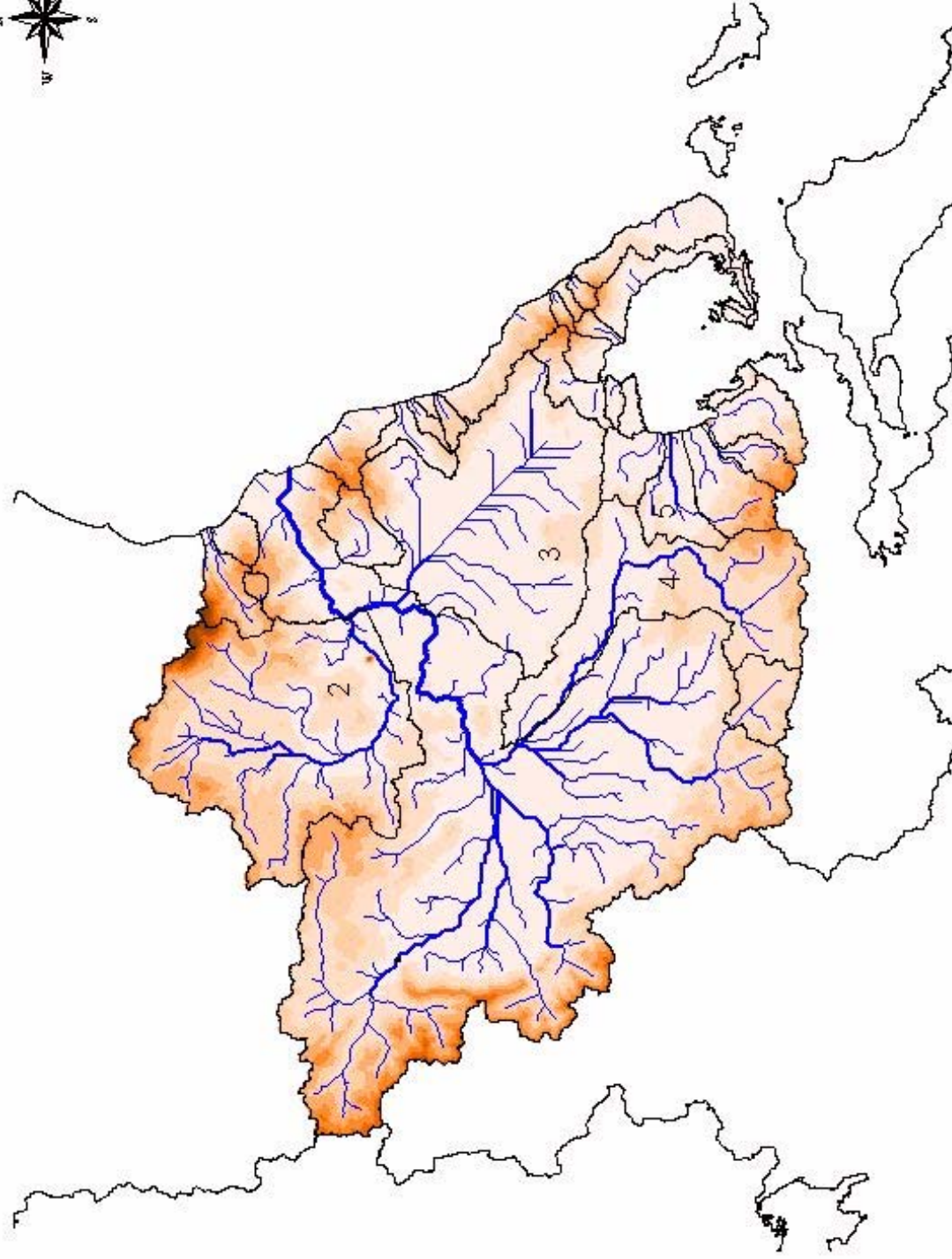
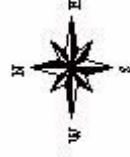
# Ανατολική Στερεά



1. Βοιωτικός Κηφισός
2. Σπερχειός
3. Ασωπός
4. Πλειστός
5. Κηρέας
6. Βισυρίτσας
7. Λήδος
8. Περιήσσου
9. Ψαγγών
10. Οξύλιθου
11. Οριότικος
12. Κάλλας
13. Κοριάκι
14. Αταλάντη
15. Πλατεονιά

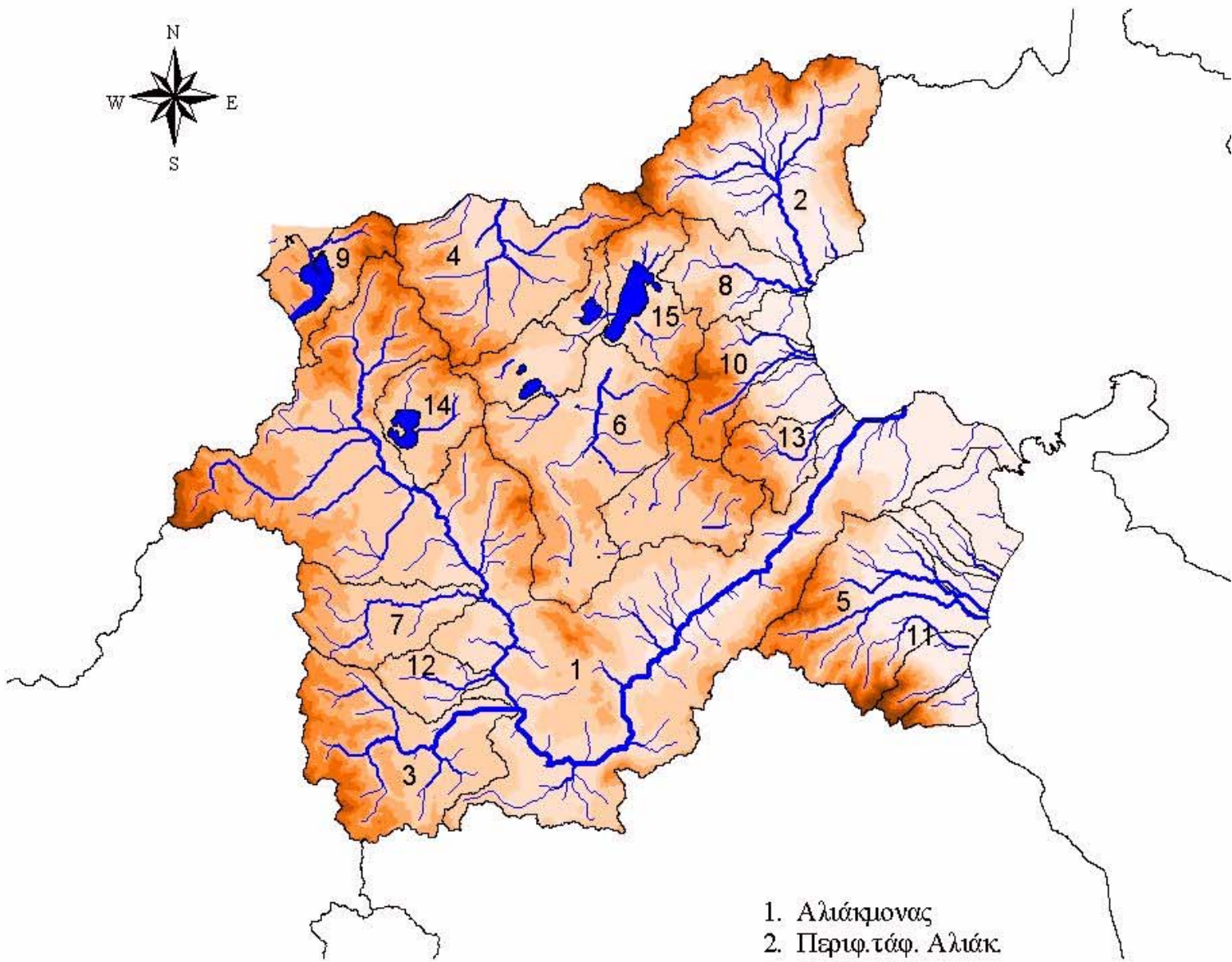


# Θεσσαλία



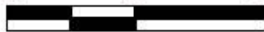
1. Πηνειός
2. Τιταρήσιος
3. Αλμυρός
4. Ενιπέας
5. Χολόρεμμα

# Δυτική Μακεδονία

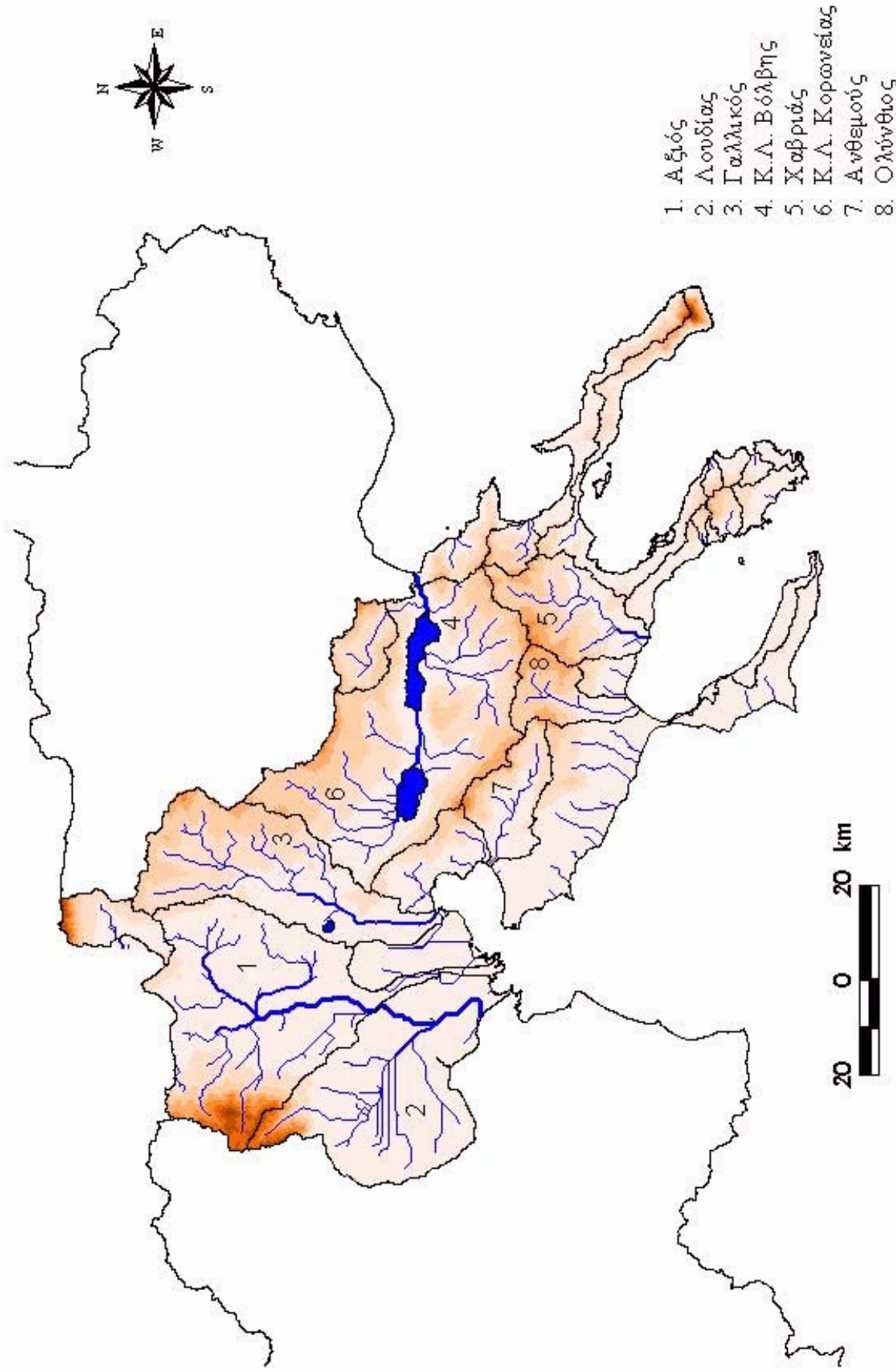


1. Αλιάκιμονας
2. Περιφ.τάφ. Αλιάκ
3. Βενέτικος
4. Αξιός (Φλωρ.)
5. Μαυρονερίου
6. Κ.Λ. Πτολεμαΐδας
7. Προμόρτσα
8. Εδεσσαίος
9. Κ.Λ. Μικρής Πρέσπας
10. Αραπίτσα
11. Χελιοπόταμος
12. Γραβενίτικος
13. Τριπόταμος
14. Κ.Λ. Καστοριάς
15. Κ.Λ. Βεγορίτιδας

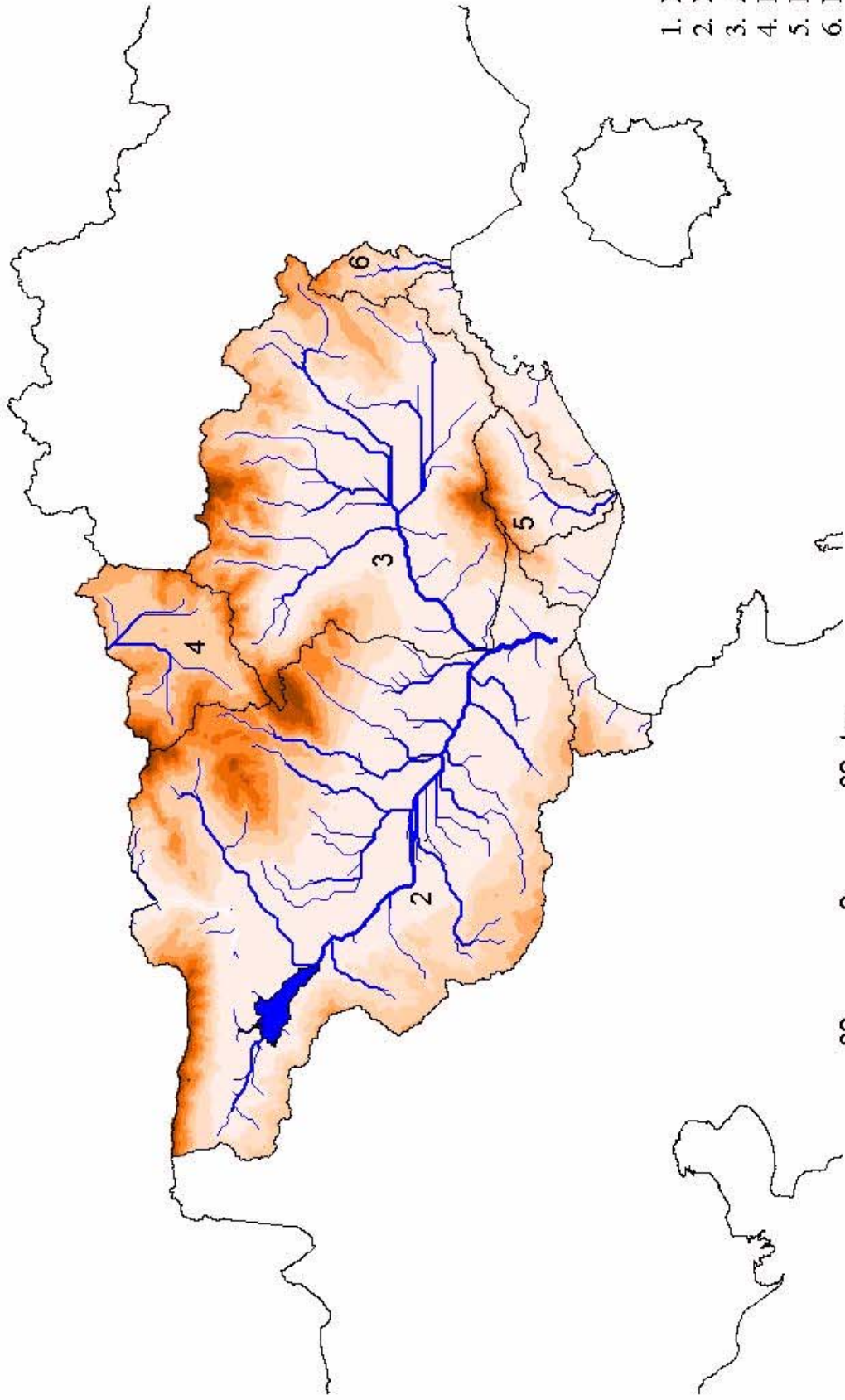
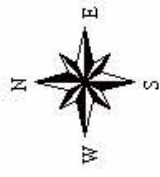
20 0 20 km



# Κεντρική Μακεδονία

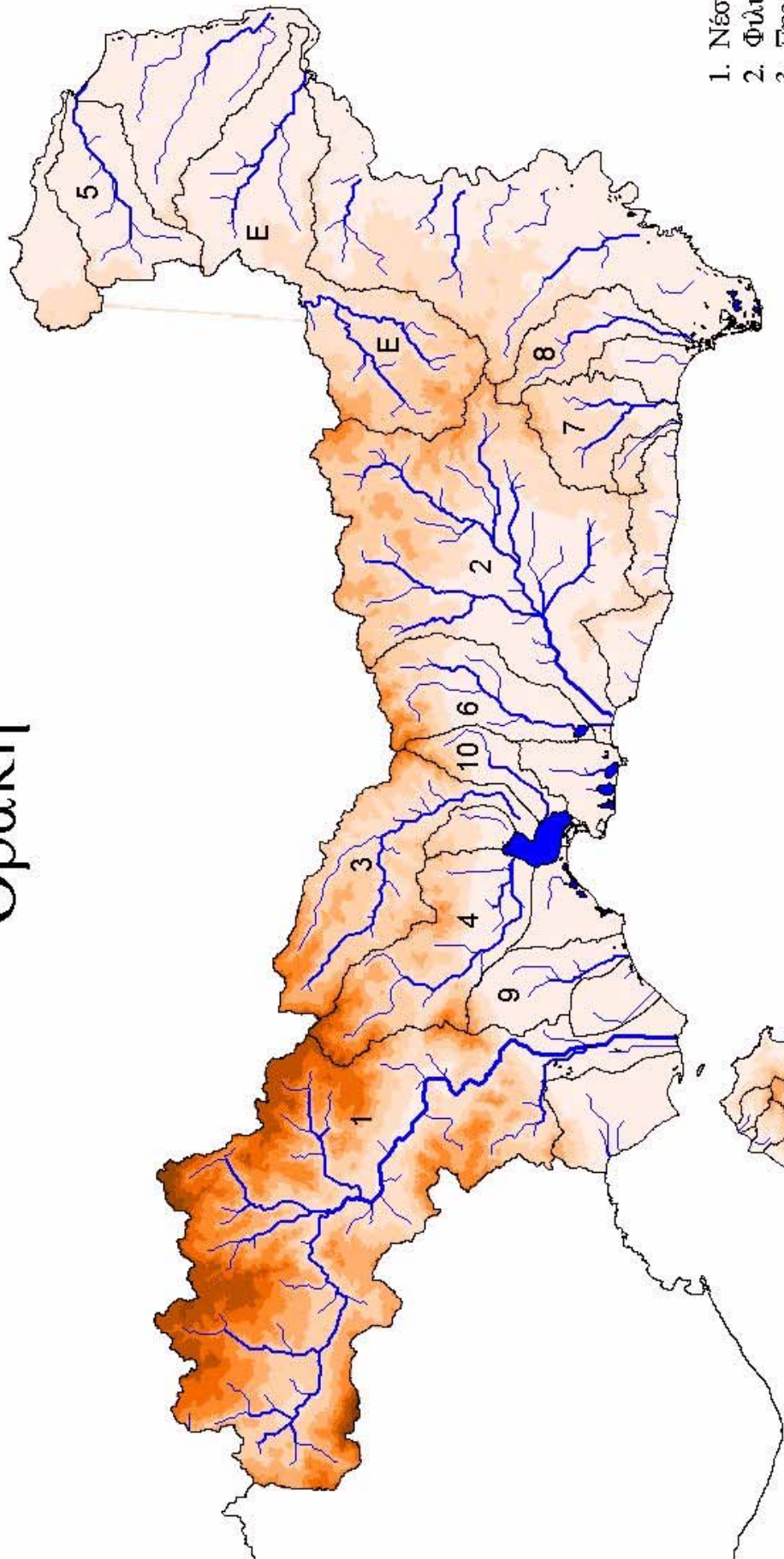
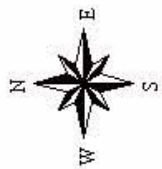


# Ανατολική Μακεδονία



1. Στρομόνας (συν)
2. Στρομόνας (υπολ.)
3. Αγγίτης
4. Κ.Α. Οχρού
5. Μερμαράς
6. Νέα Κάρβαλη

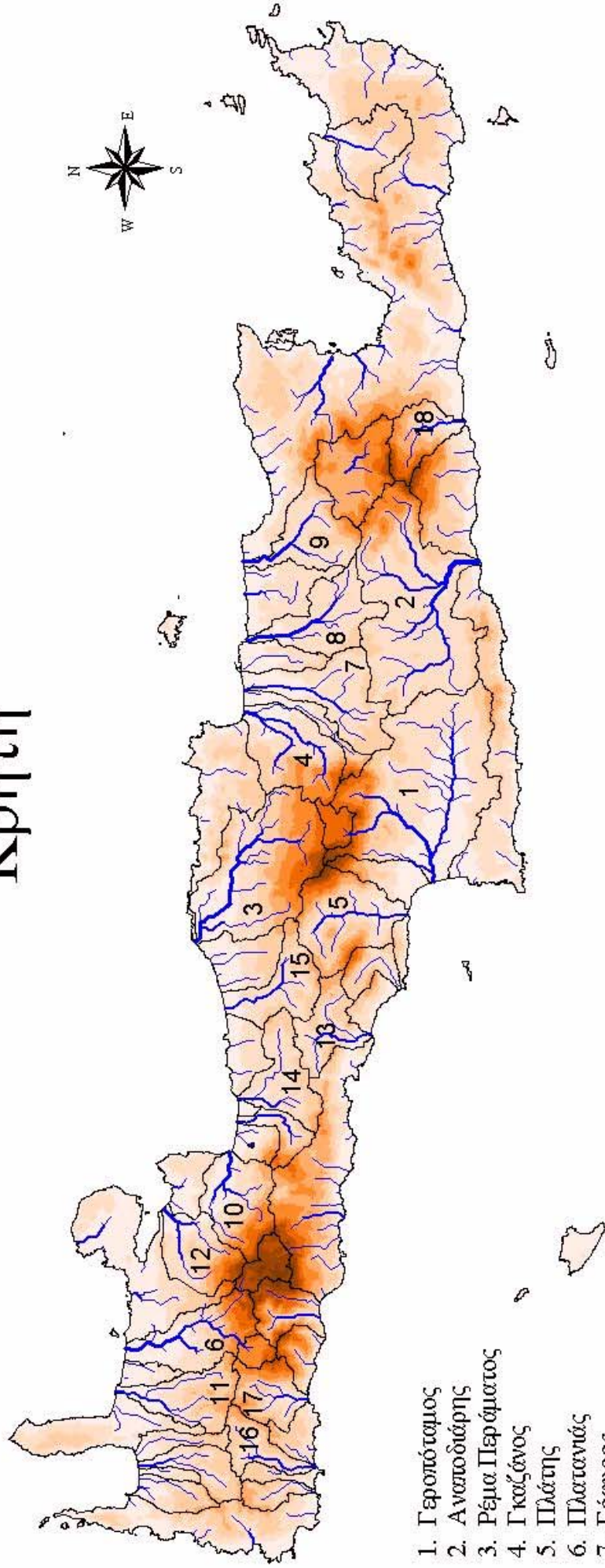
# Θράκη



1. Νέστος
  2. Φυλιούρης
  3. Εηρόρεμα
  4. Ρέμα Ειάνθης
  5. Άρδας
  6. Ρέμα Κομοτηνής
  7. Ποταμός
  8. Ρέμα Λουτρού
  9. Λασιποτάμιος
  10. Ασπροπόταμος
- Ε Ερίμανθος



# Κρήτη



1. Γεροπόταμος
2. Αναποδιάρης
3. Ρέμα Περάματος
4. Γκαζάνος
5. Πλάτης
6. Πλατανιάς
7. Γόφυρος
8. Καρτερός
9. Αποσελέμης
10. Ρέμα Βρύσες
11. Τσαρωνίτης
12. Ρέμα Καλάμι
13. Κουρταλιώτης
14. Πέρες
15. Ρέμα Πρεσιές
16. Κοκοδικιάνος
17. Σούγιας
18. Μύρτος

# **Παράρτημα Β**

## **Χαρακτηριστικά λεκανών**









# **Παράρτημα Γ**

## **Κώδικες**

## Γενικά για τη γλώσσα Avenue

Η Avenue ανήκει στις αντικειμενοστραφείς γλώσσες προγραμματισμού (object oriented). Τα πάντα στο πρόγραμμα ArcView είναι αντικείμενα τα οποία οργανώνονται σε κλάσεις και μεταξύ τους υπάρχει μια συγκεκριμένη ιεραρχία. Ο προγραμματισμός σε αντικειμενοστραφή γλώσσα απαιτεί να προσδιορίζεται σε ποια κλάση ανήκει το αντικείμενο και στη συνέχεια να στέλνεται στο αντικείμενο η εντολή μέσω της οποίας θα γίνεται μια δράση. Οι περισσότερες εντολές που χρησιμοποιούνται στην Avenue απαιτούν ορίσματα τα οποία μπαίνουν σε παρένθεση αμέσως μετά την εντολή όπως φάνηκε στην περιγραφή ορισμένων παραδειγμάτων στο 2<sup>ο</sup> κεφάλαιο.

Ας πάρουμε με τη σειρά τις εντολές του πρώτου κώδικα. Με την πρώτη γραμμή εντολής ορίζουμε ως κλάση View τον ενεργό χώρο εργασίας (ανοικτό παράθυρο στην οθόνη). Με τη δεύτερη εντολή ορίζουμε ως ElevGrid το Grid του θέματος fill1 (κάνναβος υψομέτρων) ο οποίος βρίσκεται στο χώρο εργασίας View που ορίσαμε αρχικά. Στη συνέχεια ορίζουμε ως FlowDirGrid τον κάνναβο που θα προέλθει από την εκτέλεση της εντολής FlowDirection στον κάνναβο υψομέτρων, με την παράμετρο false. Έτσι έχουμε τον κάνναβο διεύθυνσης ροής. Τον κάνναβο αυτό, θέλουμε να τον προσθέσουμε στο χώρο εργασίας View. Για γίνει αυτό, με την εντολή make δημιουργούμε ένα θεματικό επίπεδο (GTheme) το οποίο το ονομάζουμε GridTheme και στη συνέχεια προσθέτουμε το θέμα αυτό στο χώρο εργασίας. Αν θέλουμε, μπορούμε να ονομάσουμε το θέμα αυτό με την εντολή SetName, διαφορετικά λαμβάνει ένα τυχαίο όνομα. Βέβαια, το όνομα με το οποίο θα αποθηκευτεί θα είναι τυχαίο. Εμείς απλά αλλάζουμε το όνομα με το οποίο εμφανίζεται στο χώρο εργασίας View. Τα αρχεία, αποθηκεύονται στο φάκελο που έχουμε ορίσει ως φάκελο εργασίας.

### Εύρεση της διεύθυνσης ροής κάθε κυττάρου

```
theView = av.GetActiveDoc
ElevGrid = theView.FindTheme("fill1").getGrid
FlowDirGrid = theElevGrid.FlowDirection(false)
GridTheme = GTheme.Make(FlowDirGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Flow2")
```

### Εντοπισμός των κυττάρων με απροσδιόριστη διεύθυνση ροής

```
theView = av.GetActiveDoc
FlowGrid = theView.FindTheme("Flow1").GetGrid
SinkGrid = FlowGrid.Sink
gridTheme = GTheme.Make(SinkGrid)
theView.AddTheme(gridTheme)
theGTheme.SetName("Sink1")
```

## Γέμισμα των κοιλοτήτων

```
theView = av.GetActiveDoc
ElevGrid = theView.FindTheme("Fill1").GetGrid
FlowGrid = theView.FindTheme("Flow2").GetGrid
SinkGrid = theView.FindTheme("Sink2").GetGrid
waterGrid = FlowGrid.Watershed(SinkGrid)
ZonalFillGrid = waterGrid.ZonalFill(ElevGrid)
FillGrid = (ElevGrid <
    (zonalFillGrid.IsNotNull.Con(0.AsGrid, zonalFillGrid))) .Con(
    zonalFillGrid, ElevGrid)
GridTheme = GTheme.Make(FillGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Fill2")
```

## Συγκέντρωση ροής

```
theView = av.GetActiveDoc
FlowGrid = theView.FindTheme("Flow").GetGrid
flowAccumGrid = FlowGrid.FlowAccumulation(NIL)
GridTheme = GTheme.Make(flowAccumGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Accum")
```

## Δημιουργία δικτύου ποταμών και ταξινόμηση των υδατορευμάτων

```
theView = av.GetActiveDoc
AccumGrid = theView.FindTheme("Accum").GetGrid
StreamGrid = (AccumGrid < 256.AsGrid).SetNull(1.AsGrid)
FlowGrid = theView.FindTheme("Flow").GetGrid
StreamOrderGrid = StreamGrid.StreamOrder(FlowGrid, False)
GridTheme = GTheme.Make(StreamOrderGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Order256")
```

## Προετοιμασία για την εύρεση των υπολεκανών (ξεχωριστή τιμή για κάθε τμήμα)

```
theView = av.GetActiveDoc
FlowGrid = theView.FindTheme("Flow").GetGrid
OrderGrid = theView.FindTheme("Order256").GetGrid
LinkGrid = OrderGrid.StreamLink(FlowGrid)
GridTheme = GTheme.Make(LinkGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Link256")
```

## Εύρεση των υπολεκανών

```
theView = av.GetActiveDoc
FlowGrid = theView.FindTheme("Flow").GetGrid
LinkGrid = theView.FindTheme("Link256").GetGrid
WatershedGrid = FlowGrid.Watershed(LinkGrid)
GridTheme = GTheme.Make(WatershedGrid)
theView.AddTheme(GridTheme)
theGTheme.SetName("Watershed256")
```

## Μετατροπή του υδρογραφικού δικτύου σε γραμμικό θεματικό επίπεδο, διατηρώντας την τάξη κάθε τμήματος και υπολογίζοντας το μήκος τους

```
theView = av.getActiveDoc
c = theView.FindTheme("Order256").GetGrid
FlowdirGrid = theView.FindTheme("Flow").GetGrid
a = av.GetProject.GetWorkDir.MakeTmp("riv", "shp")
theFtab = c.StreamToPolyLineFtab(a,
    FlowdirGrid, True, prj.MakeNull)
TheFtab.SetEditable(True)
LengthField = TheFtab.FindField("length")
if(LengthField = nil)then
    LengthField = Field.Make("length", #FIELD_DECIMAL, 9, 0)
    TheFtab.AddFields({LengthField})
end
TheShpFld = TheFtab.FindField("Shape")
for each rec in TheFtab
    TheShpV = TheFtab.ReturnValue(TheShpFld, rec)
    TheLengV = av.run("hydro.GetPolyLineLength", TheShpV)
    TheFtab.SetValue(LengthField, rec, TheLengV)
end
theFTheme = FTheme.Make(TheFtab)
theView.AddTheme(theFTheme)
theFTheme.SetName("RiverLine256")
```

## hydro.GetPolyLineLength

Ρουτίνα υπολογισμού μήκους γραμμικών τμημάτων από την επέκταση hydro του προγράμματος ArcView.

```
arcIn=SELF
segList=arcIn.AsList
TheLength=0
for each lst in SegList
    isFirstPoint=True
```

```

SegLength=0
for each pnt in lst
  if(isFirstPoint) then
    OldPnt=Pnt
    isFirstPoint=False
  else
    SegLength=SegLength+(((Pnt.getX-
      OldPnt.getX)^2)+(Pnt.getY-
      OldPnt.getY)^2))^(0.5))
    OldPnt=Pnt.Clone
  end
end
TheLength=TheLength+SegLength
end
return (TheLength)

```

### **Εύρεση λεκάνης απορροής επιλέγοντας την έξοδό της**

Απλά αφήνουμε τον δείκτη του ποντικιού πάνω στο κύτταρο εξόδου – Υπάρχει ως παράδειγμα στο help του προγράμματος

```

theView = av.GetActiveDoc
theDisplay = theView.GetDisplay
theElevGrid = theView.FindTheme("fill").GetGrid
thePoint = theDisplay.ReturnUserPoint
mPoint = MultiPoint.Make({thePoint})
theSrcGrid =
  theElevGrid.ExtractByPoints(mPoint, Prj.MakeNull, FALSE)
theFlowDir = theView.FindTheme("Flow").GetGrid
theAccum = theView.FindTheme("Accum").GetGrid
theWater =
  theFlowDir.Watershed(theSrcGrid.SnapPourPoint(theAccum,
  100))
theGTheme = GTheme.Make(theWater)
  if (theWater.HasError) then
    return NIL
  end
theView.AddTheme(theGTheme)
theGTheme.SetName("L.Korinthou")

```

## Υπολογισμός της περιμέτρου μιας λεκάνης

Ενεργό θεματικό επίπεδο ο κάνναβος της λεκάνης που έχει παραχθεί από τον τελευταίο κώδικα. Η εντολή αυτή υπολογίζει την περίμετρο για κάθε ζώνη (στοιχεία με την ίδια τιμή στο πεδίο value) του εισαγόμενου καννάβου. – Από την επέκταση hydro του προγράμματος ArcView.

```
theView = av.GetActiveDoc
t = theView.GetActiveThemes.Get(0)
g = t.GetGrid
r = g.ZonalGeometry(#GRID_GeomDESC_PERIMETER)
aFN = av.GetProject.GetWorkDir.MakeTmp("perim", "")
r.Rename(aFN)
if (r.HasError) then return NIL end
gthm = GTheme.Make(r)
gthm.SetName("Perimeter of"++t.GetName)
theView.AddTheme(gthm)
```