

Ορίζω τον τύπο **BinTreeElementType** για τα στοιχεία του ΔΔΑ
Ορίζω τον τύπο της εγγραφής του αρχείου για τους φοιτητές

```
int BuildBST(BinTreePointer *Root);  
void PrintStudent(int RecNum);  
void printStudentsWithGrade(float theGrade);  
void writeNewStudents(BinTreePointer *Root, int *size);
```

```
int main(){  
    Δηλώνω τις μεταβλητές  
    _____ ARoot, LocPtr;  
    _____ size;           //πλήθος κόμβων του ΔΔΑ  
    _____ student;  
    _____ found;  
    _____ theGrade;  
  
    //1  
    Εμφάνισε "Q1. Build BST from a file"  
    Δημιούργησε το ευρετήριο και επέστρεψε το πλήθος των κόμβων (size=BuildBST(...))  
  
    //2  
    Εμφάνισε "Q2. Print size and BST"  
    Εμφάνισε το πλήθος των κόμβων  
    Εμφάνισε τους κόμβους του δέντρου (σε αύξουσα διάταξη)  
  
    //3  
    Εμφάνισε "Q3. Write new students, update file and BST "  
    Κάλεσε τη συνάρτηση για την εισαγωγή νέων φοιτητών (writeNewStudents(...))  
  
    //4  
    Εμφάνισε "Q4. Print size and BST"  
    Εμφάνισε το πλήθος των κόμβων  
    Εμφάνισε τους κόμβους του δέντρου (σε αύξουσα διάταξη)  
  
    //5  
    Εμφάνισε "Q5. Search for a student"  
    Διάβασε τον AM  
    Αναζήτησε τον φοιτητή με τον παραπάνω AM (RecBSTSearch(...))  
    Αν ο φοιτητής βρέθηκε  
        Εμφάνισε τα στοιχεία του φοιτητή (PrintStudent(...))  
    Αλλιώς  
        Εμφάνισε ότι ο φοιτητής με τον παραπάνω AM δεν βρέθηκε
```

//6

Εμφάνισε “Q6. Print size and BST”

Εμφάνισε το πλήθος των κόμβων

Εμφάνισε τους κόμβους του δέντρου (σε αύξουσα διάταξη)

//7

Εμφάνισε “Q7. Print students with grades >= given grade”

Διάβασε τον βαθμό theGrade

Εμφάνισε όλους τους φοιτητές με βαθμό >= theGrade (**printStudentsWithGrade(...)**)

Επέστρεψε 0;

}

```

int BuildBST(BinTreePointer *Root){
    Δηλώνω τις μεταβλητές
    _____ *fp;
    _____ nscan;
    _____ termch;
    _____ student; //εγγραφή για την ανάγνωση των στοιχείων φοιτητή από το αρχείο
    _____ indexRec; //εγγραφή φοιτητή για το ΔΔΑ
    _____ size; //πλήθος εγγραφών φοιτητή

    CreateBST(???);

    Άνοιξε το αρχείο για ανάγνωση
    Αρχικοποίησε το size
    Αν το αρχείο άνοιξε επιτυχώς {
        while (TRUE) {
            Διάβασε από το αρχείο την εγγραφή του φοιτητή
            Αν η ανάγνωση έφτασε στο τέλος του αρχείου
                break
            Αν η ανάγνωση της εγγραφής δεν ήταν επιτυχής
                Εμφάνισε λάθος
            Αλλιώς{
                Καταχώρησε στο indexRec τον AM που διάβασες
                Καταχώρησε στο indexRec τον τρέχοντα αριθμό εγγραφής
                Εισαγωγή της εγγραφής του φοιτητή (indexRec) στο ΔΔΑ
                (RecBSTInsert(...))
                Αύξησε κατά 1 το πλήθος των εγγραφών
            }
        }
        Κλείσε το αρχείο fp
        Επέστρεψε το size
    }
}

```

```

void writeNewStudents(BinTreePointer *Root, int *size){
    Δηλώνω τις μεταβλητές
    _____ *fp;
    _____ student; //εγγραφή για την ανάγνωση των στοιχείων φοιτητή από το αρχείο
    _____ indexRec; //εγγραφή φοιτητή για το ΔΔΑ
    _____ found;
    _____ LocPtr;

    Άνοιξε το αρχείο για εγγραφή στο τέλος του αρχείου
    Αν το αρχείο άνοιξε επιτυχώς {
        Επαναληπτικά{
            Διάβασε τον AM
            Αναζήτησε τον φοιτητή με τον παραπάνω AM (RecBSTSearch(...))
            Αν δε βρέθηκε φοιτητής με AM στο ΔΔΑ τότε{
                Καταχώρησε τον AM στην εγγραφή του φοιτητή για το αρχείο
                Διάβασε επώνυμο
                Διάβασε όνομα
                Διάβασε φύλο
                Διάβασε έτος
                Διάβασε βαθμό
                Καταχώρησε στον αριθμό της νέας εγγραφής του ΔΔΑ το πλήθος
                εγγραφών
                Εισαγωγή του φοιτητή στο ΔΔΑ (RecBSTInsert(...))
                Γράψε τον φοιτητή στο αρχείο (fprintf(...))
                Αυξάνω κατά 1 το πλήθος των φοιτητών
            }
            Αλλιώς
                Εμφάνισε ο AM υπάρχει ήδη στο αρχείο

            Εμφάνισε "Continue Y/N: "
            Επαναληπτικά{
                Διάβασε τον χαρακτήρα ch
                }όσο ο χαρακτήρας δεν είναι N ή Y //μικρά ή κεφαλαία
            }όσο ο χαρακτήρας δεν είναι N
            Κλείσε το αρχείο
        }
    }
}

```

```

void PrintStudent(int recNum){
    Δηλώνω τις μεταβλητές
    _____ *fp;
    _____ nscan;
    _____ termch;
    _____ student; //εγγραφή για την ανάγνωση των στοιχείων φοιτητή από το αρχείο
    _____ indexRec; //εγγραφή φοιτητή για το ΔΔΑ
    _____ lines; //γραμμές του αρχείου που διαβάστηκαν

    Άνοιξε το αρχείο για ανάγνωση
    Αρχικοποίησε το lines
    Αν το αρχείο άνοιξε επιτυχώς {
        Όσο δεν έφτασε στη γραμμή του ζητούμενου φοιτητή {
            Διάβασε από το αρχείο την εγγραφή του φοιτητή
            Αν η ανάγνωση έφτασε στο τέλος του αρχείου
                break
            Αν η ανάγνωση της εγγραφής δεν ήταν επιτυχής
                Εμφάνισε λάθος
            Αύξησε το πλήθος των γραμμών
        }
        Αν οι γραμμές που διαβάστηκαν είναι ίσες με τον αριθμό της εγγραφής του
        φοιτητή + 1
            Εμφάνισε όλα τα στοιχεία του φοιτητή
        }
    Κλείσε το αρχείο
}

```

lines

rec = 8

```

void printStudentsWithGrade (float theGrade)
{
    Δηλώνω τις μεταβλητές
    _____ *fp;
    _____ nscan;
    _____ termch;
    _____ student;

    Άνοιξε το αρχείο για ανάγνωση
    Αν το αρχείο άνοιξε επιτυχώς {
        while (TRUE) {
            Διάβασε από το αρχείο την εγγραφή του φοιτητή
            Αν η ανάγνωση έφτασε στο τέλος του αρχείου
                break
            Αν η ανάγνωση της εγγραφής δεν ήταν επιτυχής
                Εμφάνισε λάθος
            Αλλιώς
                Αν ο βαθμός του φοιτητή >= theGrade
                    Εμφάνισε τα στοιχεία του φοιτητή
            }
            Κλείσε το αρχείο
        }
    }
}

```