

Using arrays, files and strings:

Reading grades from a file and processing the data:

1. Ask the user for the name of the file containing grades.
2. Call the function to read the file.
3. Call the function to find the topper of the class.
4. Call the function to find the average grade of the class.
5. Print the points obtained by the topper and the class average.

```
#include <stdio.h>
#define SIZE 10

void readfile(char filename[], int grade[], int n);
double perform ( int grade[], int n);
int topper ( int A[ ], int nn);

int main(void) {
    int grade[SIZE],topper_position;
    double av;
    char filename[30];
    printf("\nenter the name of the file.\n");
    scanf("%s",filename);
    readfile(filename, grade,SIZE);
    topper_position = topper( grade , SIZE);
    av = perform( grade, SIZE);
    printf("\ntopper got %d points.",
           grade[topper_position]);
    printf("\nclass average is %.2f points.",av);
    return 0;
}

void readfile(char filename[], int grade[], int n)
{
    int size, i;
    FILE *ifp;
    ifp = fopen(filename, "r");
    for (i=0; i< n; i++)
        fscanf(ifp, "%d", &grade[i] );
    fclose(ifp);
}

double perform ( int grade[], int n)
{
    int total= 0, i;
    double average;
    for (i=0; i < n; i++)
```

```
        total = total+ grade[i];
average = total/10.0;
return average;
}
```

```
int topper ( int A[ ], int nn)
{
    int i, max_pos;
    max_pos = nn-1;
    for (i=0; i < nn-1; i++)
        if ( A[i]> A[max_pos] )
            max_pos = i;
    return max_pos;
}
```

Exam.txt

30
40
40
50
50
60
60
90
70
60

topper got 90 points.
class average is 55.00 points.

Reading names of students and their grades from a file and processing the data:

1. Ask the user for the name of the file containing grades.
2. Call the function to read the name of each student and corresponding grade.
3. Call the function to find the position of the topper of the class.
4. Call the function to find the average grade of the class.
5. Call the print function to print the names of the students and points obtained by them, and the name of the topper and the points obtained.

```
#include <stdio.h>
#define SIZE 10

void readfile(char filename[], char student[][30], int
              grade[], int n);

int topper ( int A[ ], int nn);
void printing (char student[][30],grade[],int n, int
              top_pos)

int main(void) {
    int grade[SIZE],topper_position, i;
    double av;
    char filename[30], student[SIZE][30];
    printf("\nenter the name of the file.\n");
    scanf("%s",filename);
    readfile(filename, student, grade,SIZE);
        topper_position = topper( grade , SIZE);
    av = perform( grade, SIZE);
    printing (student, grade, SIZE, top_pos);

    return 0;
}

void readfile(char filename[], char student[][30], int
              grade[], int n)
{
    int size, i;
    FILE *ifp;
    ifp = fopen(filename, "r");
    for (i=0; i< n; i++) {
        fscanf(ifp, "%s", student[i] );
        fscanf(ifp, "%d", &grade[i] );
    }
    fclose(ifp);
}
```

```

void printing (char student[][30],grade[],int n, int
               top_pos)
{
    int i;
    for (i=0; i< n; i++)
        printf("\n%s \t %d", student[i] , grade[i] );
    printf("\n%s is the topper and he got %d points.",
           student [top_pos],grade[top_pos]);
    printf("\nclass average is %.2f points.",av);
}

```

Examdata.txt

Sonya 30
 David 40
 Denise 40
 Julie 50
 Ashley 50
 Ron 60
 Stanley 60
 Kegan 90
 Sam 70
 John 60

Sonya 30
David 40
Denise 40
Julie 50
Ashley 50
Ron 60
Stanley 60
Kegan 90
Sam 70
John 60

Kegan is the topper and he got 90 points.
class average is 55.00 points.

Reading names of students and to enquire about the grade of a specific student.

1. Ask the user for the name of the file containing grades.
2. Call the function to read the name of each student and corresponding grade.
3. Ask the user for the name of the student whose grade is needed.
4. If the name is found, print the grade, else print "nobody by that name".

```
#include <stdio.h>
#define SIZE 10

void readfile(char filename[], char student[][30], int
              grade[], int n);
int find( char student[][30], char name[], int n);

int main(void) {
    int grade[SIZE],topper_position, i, newpos;
    double av;
    char filename[30], newname[30], student[SIZE][30];
    printf("\nenter the name of the file.\n");
    scanf("%s",filename);
    readfile(filename, student, grade,SIZE);
    for (i=0; i< SIZE; i++)
        printf("\n%s \t %d", student[i] , grade[i]
);

    printf("\nwhose grade do you want?\n");
    scanf("%s",newname);
    newpos = find(student, newname,SIZE);

    if ( newpos == -1)
        printf("\n There is no student by this name.");
    else
        printf("\n%s has got %d points.", newname,
              grade[newpos]);
    return 0;
}

int find( char student[][30], char name[], int n)
{
    int i;
    for (i=0; i < n; i++)
        if(strcmp(name,student[i])==0)
            return i;
    return -1;
}
```

Sonya 30

David 40

Denise 40

Julie 50

Ashley 50

Ron 60

Stanley 60

Kegan 90

Sam 70

John 60

whose grade do you want?

Ashley

Ashley has got 50 points.

Reading names of students and to replace one of the names.

1. Ask the user for the name in the list.
2. If found, string copy the new name(say Joan) on the old name.
3. if not found, report it.

```
printf("\nwhich name you want to change?");
scanf("%s",newname);
newpos = find(student, grade, newname,SIZE);
printf("\nnewpos= %d",newpos);
if ( newpos == -1)
    printf("\nThere is no student by this name.");
else
    strcpy(student[newpos],"Joan");
```

examdata.txt

```
Sonya
David
Denise
Julie
Ashley
Ron
Stan
Kegan
Sam
John
```

which name you want to change?

Ashley

```
Sonya
David
Denise
Julie
Joan
Ron
Stan
Kegan
Sam
John
```

Reversing the order of elements in a string array.

1. Use a for loop to go to middle of the array.
2. Swap the last name with the first name, that is, swap the item located at position 0 with item located at n-1, and in general, swap item at i with item located at n-1-i.

The contents of an array `item_list[5]` is shown below.

```
item_list[5]  
coffee  
bread  
salt  
milk  
pepsi
```

Write a code to reverse the contents of the list. Make it general enough so that it can handle n items in the array.

```
char item_list[50][20], temp[20];  
for( i = 0; i < n/2; i++) {  
    strcpy(temp, item_list[i] );  
    strcpy(item_list[i], item_list[n-1-i] );  
    strcpy(item_list[n-1-i], temp );  
}
```

The contents of an array `item_list[5]` is shown below.

```
item_list[5]  
coffee  
bread  
salt  
milk  
pepsi
```

After reversal it has the following form:

```
Pepsi  
milk  
salt  
bread  
coffee
```

