You are to make minimal modifications to in-class sample program, grades4 in Pointers 4, to use the following derived type in place of int grades:

```c
struct Student
{
    char* pname;
    int grade;
};
```

Requirements:
1. Include a header comment that gives the following information:

```c
// Project name
// your last name, first name
// 110B - your section number
```

// Brief statement of the program's task

2. Your program should consist of main() and six functions. The dynamic array will be Student type rather than int. Note that the student's name will be stored in the free store, and struct member, pname, will point to the name. Identifiers should not refer to grades but to Students. The format of the output must be modified to display the names as well as the grades. The tasks performed by the various functions will remain basically the same.

3. Use good programming style, that is, vertically align matching braces, code nothing but a brace on a source line, use proper indentation, and use whitespace judiciously.

4. Generally, format your report as follows. Run your program 5 times using the following test data which is shown in bold.

Run 1:
Enter the maximum number of students=>5
Enter a maximum of 5 students.
Enter the name of the first student (<Enter> to QUIT): Joe E. Black
Enter student's grade: 75
Enter the name of the next student (<Enter> to QUIT): Pat Wong
Enter student's grade: 80
Enter the name of the next student (<Enter> to QUIT): Jim Garcia
Enter student's grade: 99
Enter the name of the next student (<Enter> to QUIT): Sue Tang
Enter student's grade: 100
Enter the name of the next student (<Enter> to QUIT): Sam Hadwan
Enter student's grade: 74

Your 5 students are:

Joe E. Black    75
Pat Wong        80
Jim Garcia      99
Sue Tang        100
Sam Hadwan      74

Class average is: 85.6
Run 2:
Enter the maximum number of students=>5
Enter a maximum of 5 students.
Enter the name of the first student (<Enter> to QUIT): Joe Edward Black
Enter student's grade: 75
Enter the name of the next student (<Enter> to QUIT): Pat Wong
Enter student's grade: 80
Enter the name of the next student (<Enter> to QUIT): James Garcia
Enter student's grade: 99
Enter the name of the next student (<Enter> to QUIT):

Your 3 students are:

Joe Edward Black 75
Pat Wong 80
James Garcia 99

Class average is: 84.7

Run 3:
Enter the maximum number of students=>1
Enter a maximum of 1 students.
Enter the name of the first student (<Enter> to QUIT): James F. Garcia
Enter student's grade: 100

Your 1 student is:

James F. Garcia 100

Class average is: 100.0

Run 4:
Enter the maximum number of students=>5
Enter a maximum of 5 students.
Enter the name of the first student (<Enter> to QUIT):

You entered no students.

Run 5:
Enter the maximum number of students=>0

Maximum number of students must be positive.

Hint:
1. See sample program, RagArrNames in C Stings2, for setting up a string in the free store.
2. See sample program, structArray in Structures, for a data entry loop.
Dynamic Array for Run 2:

- **pStu + 0**
  - `pname`: Joe\^Edward\^Black\0
  - `grade`: 75

- **pStu + 1**
  - `pname`: Pat\^Wong\0
  - `grade`: 80

- **pStu + 2**
  - `pname`: James\^Garcia\0
  - `grade`: 99

### Syntax to access members

1. `pStu[i].pname`
2. `(pStu + i)->pname`
3. `(*(pStu + i)).pname`