1. When the following C++ code is executed, the output is the hexadecimal number 0xbffff854.
   (a) Just before the program terminates, what is the contents of \( j \)?
   (b) Just before the program terminates, what is the contents of \( jp \)?

```cpp
#include <iostream>
using namespace std;

int main()
{
    int j = 16;
    int* jp;
    cout << &j;
    jp = &j;
    return 0;
}
```

2. Consider the following array declarations:

```cpp
int a[4] = {12, 24, 36, 48};
int* p[4];
```
   (a) Write correct C++ code to assign the address of \( a[i] \) to \( p[i] \) for \( i = 0, \ldots, 4 \).
   (b) Using the "boxes and arrows" technique demonstrated in class, show the result of executing the code described in part 2a.
   (c) Write correct C++ code to modify array p so that \( p[2] \) points to \( a[1] \) and \( p[1] \) points to \( a[2] \).
   (d) Using the "boxes and arrows" technique demonstrated in class, show the result of executing the code described in part 2c.

   (a) Write a correct C++ definition for a struct called Contacts that contains the following data:
      - an array of characters of length MAXNAME. This array should be called name.
      - an array of integers of length MAXPHONE. This array should be called phone.
      - an integer called zip.
   You should assume that MAXNAME and MAXPHONE are constant integers that have been defined.
   (b) Write correct C++ code to print to the screen the contents of zip.
   (c) Write correct C++ code to set the value of zip to be 78666.
   (d) Write correct C++ code to set the value of name to be "Turing".
   (e) Write correct C++ code to declare an array of Contacts of size 50.
   (f) Write correct C++ code to declare a function called getInitial that returns the first character in the name. You need to write the function header (return type, function name, and parameter list), but not the implementation. Briefly explain how the function accesses the data in the struct.
4. Consider the following class definition:

```cpp
#include <iostream>
#include <stdlib.h>
using namespace std;

class Grades
{
  public:
    Grades();
    Grades(char*);
    void setsemester(char *sem);
    char *getsemester();
    void setmath(char g);
    char getmath();
    void setcs(char g);
    char getcs();
    friend ostream &operator << (ostream&os, const Grades & gr)
    {
      os << "semester: " << gr.semester << endl
         << "math: " << gr.math << endl
         << " cs: " << gr.cs << endl;
      return os;
    }

  private:
    char math; // letter grade
    char cs; // letter grade
    char *semester; // name of semester
};
```

(a) Declare an instance of this class called grF08.
(b) Declare a variable called grptr that is a pointer of type Grades, that is, the variable grptr points to an instance of class Grades.
(c) Identify each of the following features in the class definition:
   i. constructors
   ii. default constructors
   iii. accessors
   iv. mutators
   v. overloaded operators
(d) Why is the keyword friend needed?
(e) Write a correct C++ implementation for setmath.
(f) If the class definition is in a file called grades.h, in what file should the function implementation from part 4e be placed?