

2. (20 points) Give the output for the following program. Be sure to note parameters and transmission modes for all functions.

```
1 #include <iostream>
2 class Pokemon {
3 public:
4     Pokemon( ) : combatPower(0) {
5         std::cout << "default" << std::endl;
6     }
7     Pokemon(int c) : combatPower(c) {
8         std::cout << "conversion" << std::endl;
9     }
10    Pokemon(const Pokemon& p) :
11        combatPower(p.combatPower) {
12        std::cout << "copy" << std::endl;
13    }
14    Pokemon& operator=(const Pokemon&) {
15        std::cout << "assignment" << std::endl;
16        return *this;
17    }
18    unsigned int getCP() const { return combatPower; }
19 private:
20    unsigned int combatPower;
21 };
22 void display(const Pokemon pokemon) {
23     std::cout << pokemon.getCP() << std::endl;
24 }
25 int main() {
26     Pokemon snorlax(2840), dragonite = snorlax;
27     display(dragonite);
28 }
```

-
3. (20 points) The program below ends with a double free error. Write one function to fix the problem.

```
1 #include <cstring>
2 #include <iostream>
3 class string {
4 public:
5     string(const char* s) : buf(new char[strlen(s)+1]) { strcpy(buf, s); }
6     ~string() { delete [] buf; }
7     const char* getBuf() const { return buf; }
8 private:
9     char * buf;
10 };
11
12 int main() {
13     string a("cat"), b = a;
14 }
```

```

1  #include <cstring>
2  #include <iostream>
3
4  class Pokemon {
5  public:
6      Pokemon(const char* n, int cp) :
7          name(new char[strlen(n)+1]), combatPower(cp) {
8          strcpy(name, n);
9      }
10     unsigned int getCP() const { return combatPower; }
11 private:
12     char* name;
13     unsigned int combatPower;
14 };
15
16 int main() {
17     Pokemon snorlax("snore", 2850), dragonite = snorlax;
18     std::cout << dragonite << std::endl;
19 }

```

4. (40 points)

- (a) Write a `getName()` function and an output operator for class `Pokemon` above.
- (b) Write a function `setName(const char* n)`, which changes the name of the pokemon to `n`.
- (c) Write an *assignment operator* for class `Pokemon`.
- (d) Explicitly disallow the use of compiler generated constructors.