COMP2012 Object-Oriented Programming and Data Structures
Practice Problems on Pointers and Memory Allocation

The objective of this set of practice problems is to train your concepts on pointers, arrays, static and dynamic allocation.

Questions

State the problem(s) for the code in question 1 - 6. Also, suggest how to fix the problems.

1.
```
#include <iostream>
using namespace std;
int main( ) {
    int i=1, j=2, k=3, n=4, m=5;
    int *x, *y, *z;
    *x = &i;
    cout << "x: " << x << endl;
    cout << "*x: " << *x << endl;
    cout << "&x: " << &x << endl;
    cout << "i: " << i << endl;
    cout << "*i: " << *i << endl;
    cout << "&i: " << &i << endl;
    return 0;
}
```

2.
```
#include <iostream>
using namespace std;
int main() {
    int x = 2;
    int* px;
    *px = &x;
    cout << *px << endl;
    return 0;
}
```

3.
```
#include <iostream>
using namespace std;
int main() {
    int x = 2, y = 10;
    int* px, py;
    px = &x;
    py = &y;
    cout << *px << endl;
    cout << *py << endl;
    return 0;
}
```


4.
```
#include <iostream>
using namespace std;
int main() {
    int x = 10;
    int* px = &x;
    int* py = &px;
    cout << "x: " << x << endl;
    cout << "*px: " << *px << endl;
    cout << "*py: " << *py << endl;
    return 0;
}
```

5.
```
#include <iostream>
using namespace std;
int main() {
    int i=1, j=5, k=7, n=9, m=18;
    int *x, *y, *z;
    x = &i;
    y = z = &j;
    z = x;
    *z = 8;
    y = &k;
    x = &y;
    n = 5;
    *x = 4;
    cout << "i: " << i << endl;
    cout << "j: " << j << endl;
    cout << "k: " << k << endl;
    cout << "n: " << n << endl;
    cout << "m: " << m << endl;
    cout << "*x: " << *x << endl;
    cout << "*y: " << *y << endl;
    cout << "*z: " << *z << endl;
    return 0;
}
```

6.
```
#include <iostream>
using namespace std;
int main() {
    int i=1, j=2, k=3, n=4, m=5;
    int *x, *y, *z;
    x = &i;
    y = z = &j;
    z = x;
    char c = 'c';
    const char d = 'd';
    char * const ptr1 = &c;
    ptr1 = &d;
    const char *ptr2 = &d;
    *ptr2 = 'e';
    return 0;
}
```

7. Consider two pointers xp1 and xp2, which are defined as follows:

```
int x = 20, y = 30;
const int* xp1 = &x;
int* const xp2 = &y;
```

Which of the following statements is / are not valid?

a) `xp1 = &y;`
b) `xp2 = xp1;`
c) `*xp1 += *xp2;`
d) `*xp2 += *xp1;`

8. Consider the following code:

```
int main() {
    int x = 20;
    int& y = x;
    int* px = &x;
    int** ppx = &px;
    int*& py = px;
    x++;
    return 0;
}
```

a) Describe each of the variables y, px, ppx, and py in relation to variable x.
b) Use each of the variables y, px, ppx, and py to write an equivalent statement to x++;

9. What is the output of the following program?

```
#include <iostream>
using namespace std;
int main() {
    int a = 5, b = 8, c = 10;
    int* x = &a;
    int* y;
    int* z = &b;
    *x = 12;
    ++(*x);
    y = x = &c;
    c++;
    z = &a;
    --a;
    (*z)++;
    *y = 6;
    cout << a << " " << b << " " << c << endl;
    cout << *x << " " << *y << " " << *z << endl;
    return 0;
}
```


10. What is the output of the following program?

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

int* func(int* x) {
    *x *= 2;
    *x += 1;
    return x;
}

int main() {
    int m, n=10;
    int* p;
    int* q = &n;
    ++(*q);
    m = *q;
    m++;
    p = func(q);
    *p -= 5;
    *q += 5;
    cout << n << " " << m << endl;
    cout << *p << " " << *q << endl;
    return 0;
}
```

11. What is the output of the following program?

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

int main() {
    int a = 9, b = 5, c = 3;
    int* s = &a;
    int* p = &b;
    int* r = &c;
    (*p)--;
    p = s;
    *p = 2;
    *r = *p + *s + 2;
    s = r;
    r = &b;
    cout << a << b << c << endl;
    cout << *p << *r << *s << endl;
    return 0;
}
```

12. What is the output of the following program?

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

void fun(int* a, int* b, int c, int& d) {
    *a = *b;
    (*b)--;
    c += 2;
    d *= 3;
}

int main() {
    int x = 1, y = 3, z = 4, w = 2;
    fun(&x, &y, z, w);
    cout << x << y << z << w << endl;
    return 0;
}
```

13. a) What is the output produced by the following code?

```
int a = 1;
int b = 2;
int* p1, *p2;
p1 = &a;
p2 = &b;
*p1 = 10;
*p2 = 20;
cout << *p1 << " " << *p2 << endl;
p1 = p2;
cout << *p1 << " " << *p2 << endl;
*p1 = 30;
cout << *p1 << " " << *p2 << endl;
```

b) How would the output change if you were to replace

```
*p1 = 30;
```

with the following

```
*p2 = 30;
```


14. What is the output produced by the following code?

```
int a = 1;
int b = 2;
int* p1, *p2;
p1 = &a;
p2 = &b;
*p1 = 10;
*p2 = 20;
cout << *p1 << " " << *p2 << endl;
*p1 = *p2;
cout << *p1 << " " << *p2 << endl;
*p1 = 30;
cout << *p1 << " " << *p2 << endl;
```

15. What is the output of the following code segment? The code is assumed to be embedded in a correct and complete program.

```
int a[10];
int* p = &a[1];
for(int i=0; i<10; i++)
    a[i] = i;
for(int i=0; i<9; i++)
    cout << p[i] << " ";
cout << endl;
```

16. What is the output of the following code segment? The code is assumed to be embedded in a correct and complete program?

```
int array_size = 10;
int* a = new int[array_size];
int* p = a;
for(int i=0; i<array_size; i++)
    a[i] = i;
p[0] = 10;
for(int i=0; i<array_size; i++)
    cout << a[i] << " ";
cout << endl;
delete [] a;
```

17. What is the output of the following code segment? The code is assumed to be embedded in a correct and complete program.

```
int array_size = 10;
int* a = new int[array_size];
for(int i=0; i<array_size; i++)
    a[i] = 0;
for(int i=0; i<array_size; i+=2)
    *(a+i) = i;
for(int i=0; i<array_size; i++)
    cout << a[i] << " ";
cout << endl;
delete [] a;
```

18. What is the output of the following code segment? The code is assumed to be embedded in a correct and complete program.

```
int array_size = 10;
int* a = new int[array_size];
for(int i=0; i<array_size; i++)
    a[i] = i;
while(*a < 9) {
    a++;
    cout << *a << " ";
}
cout << endl;
delete [] a;
```

19. Suppose a dynamic variable were created as follows:

```
char* p;
p = new char;
```

Assuming that the value of the pointer variable p has not changed (so it still points to the same dynamic variable), how can you destroy this new dynamic variable and return the memory it uses to the operating system so that the memory can be reused.

20. For each of the problems that follow, give a memory snapshot that includes all objects after the problem statements are executed. Include as much information as possible. Use question marks to indicate memory locations that have not been initialized.

a)
```
double x = 15.6;
double y = 10.2;
double* ptr_1 = &y;
double* ptr_2 = &x;
*ptr_1 = *ptr_2 + x;
```

b)
```
int w = 10;
int x = 2;
int* ptr_2 = &x;
*ptr_2 -= w;
```

c)
```
int x[5] = { 2, 4, 6, 8, 3 };
int* ptr_1 = nullptr;
int* ptr_2 = nullptr;
int* ptr_3 = nullptr;
ptr_3 = &x[0];
ptr_1 = ptr_3 + 2;
ptr_2 = ptr_3 + 2;
```

d)
```
int w[4];
int* first_ptr = nullptr;
int* last_ptr = nullptr;
first_ptr = w;
last_ptr = first_ptr + 3;
```

21. Assume that an array g is defined with the following statement:

```
int g[] = { 2, 4, 5, 8, 10, 32, 78 };
int* ptr1 = g;
int* ptr2 = &g[3];
```

Give a memory snapshot, including the array values. Using this information, give the value of the following statements.

a) `*g`
b) `*(g+1)`
c) `*g + 1`
d) `*(g + 5)`
e) `*ptr1`
f) `*ptr2`
g) `*(ptr1 + 1)`
h) `*(ptr2 + 2)`

22. Assume that following variables have been defined with the following statements:

```
int i = 5;
int j = 10;
int* iPtr = &i;
const int* cPtr = &j;
int* const jPtr = &j;
```

Determine whether the following statements are valid or invalid.

a) `cPtr = jPtr;`
b) `jPtr = cPtr;`
c) `*cPtr = *jPtr;`
d) `*cPtr = *iPtr;`
e) `*jPtr = *cPtr;`
f) `iPtr = cPtr;`

23. Give the corresponding snapshot of memory after the following set of statements is executed:

```
double name;
double x = 20.5;
double* a = &x;
name = *a;
```

Assume the address of name is 10 and the address of x is 14.

24. Given the following statements:

```
int i1, i2;
int* p1, *p2;
i1 = 5;
p1 = &i1;
i2 = *p1/2 + 10;
p2 = p1;
```

a) What is the value of i1?
b) What is the value of i2?
c) What is the value of *p1?
d) What is the value of *p2?

25. What unfortunate misinterpretation can occur with the following declaration?

```
int* int_ptr1, int_ptr2;
```

26. Give three uses of the * operator. State what the * is doing, and name the use of the * that you present.

27. What are the two common pointer errors made by many C++ programmers? Present an example for each of these common errors.

28. What is the output of the following program?

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

void figure_me_out(int& x, int y, int& z);

int main() {
    int a, b, c;
    a = 10;
    b = 20;
    c = 30;
    figure_me_out(a, b, c);
    cout << a << " " << b << " " << c;
    return 0;
}

void figure_me_out(int& x, int y, int& z) {
    cout << x << " " << y << " " << z << endl;
    x = 1;
    y = 2;
    z = 3;
    cout << x << " " << y << " " << z << endl;
}
```