

5. (15 points) Fill out empty box below with the output of following C++ code.

```
#include <iostream>
#include <stdio.h>
int main()
{
    int x = 3;
    int y = 7;
    int &z = y;
    int &u = x;
    int v = 11;
    u = v;
    v++;
    u=u-2;
    z=x;
    --x;
    y++;
    printf("x=%d, y=%d, z=%d, u=%d, v=%d\n",x,y,z,u,v);
    return 0;
}
```

output :

6. (15 points) Read the C++ code, comments, and output of the following program and fill out empty boxes with appropriate code.

```
#include <stdio.h>

void square(  )
{

}

int main()
{
    int x=5;
    square(  ); // computes  $x^2$  and the result is assigned to x
    printf("x=%d\n",x); // x=25 will be printed because 25 is  $5^2$ 
    return 0;
}
```

Output :

x=25

7. (16 points) C++ code below shows generic stack implementation using template. Fill out empty boxes with appropriate codes.

```
template<typename T>
class Stack {
    int size;
    int top;
    T *stackPtr;
public:
    Stack(int n) { size=n; top=0; stackPtr=new T[size]; }
    ~Stack() { delete[] stackPtr; }
    bool push(  ); // return true if push is successful
                                     // return false if the stack is full
    bool pop(  ); // return true if pop is successful
                                     // return false if the stack is empty
    bool isEmpty() { if (top<=0) return true; else return false; }
    bool isFull() { if (top>=size) return true; else return false; }
};
```

```
#include <iostream>
int main()
{
    int x, y;
    float xf, yf;
    Stack<int> s1(5);
    Stack<float> s2(5);
    s1.push(5); s1.push(8);
    s1.pop(x); s1.pop(y);
    s2.push(5.3); s2.push(8.1);
    s2.pop(xf); s2.pop(yf);
    std::cout <<x<< " " <<y<< std::endl;
    std::cout <<xf<< " " <<yf<< std::endl;

    return 0;
}
```

// Insert your code for push member function

// Insert your code for pop member function

Output :

8 5
8.1 5.3