

1. What does the following print? (2 points each)

```
char *a[4]={"this", "is", "a", "test"};
char *p = a[3];

cout << 6 + 20 % 6;
cout << 1/5 + 0.5;
cout << char(3 + 'b');
cout << string(4, 'a');
cout << string("bcd")+ "a";

cout << p;
cout << a[1];
cout << p+1;
cout << *(p-2);
cout << *p--;
```

**ANSWERS**

```
8
0.5
e
aaaa
bcda

test
is
est
(undefined, free question)
t
```

2. Write a program that reads a list of words (separated by white space) until end of file, then prints on separate lines all of the words whose length is equal to exactly half the length of the longest word. If the longest word has odd length, then don't print anything. For example (30 points)

```
this is a test      (user input)
here it is too
^Z                  (Ctrl-Z = end of file)
is                  (your program output)
it
is
```

**// ANSWER**

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

int main()
{
    // Save list of words and find longest
    vector<string> list;
    string word;
    int longest = 0;
    while (cin >> word)
    {
        list.push_back(word);
        if (int(word.size()) > longest)
            longest = int(word.size());
    }

    // Print words 1/2 the longest word
    for (int i=0; i<int(list.size()); ++i)
        if (int(list[i].size())*2 == longest)
            cout << list[i] << "\n";
    return 0;
}
```

3. Write a function **equalDigits** that takes an int (you may assume between 0 and 1000000000 inclusive) and returns true if and only if all the digits are the same. For example, it would return true if passed 3333333, 77, or 9, or return false if passed 223, 1000, or 88886. (25 points)

**// ANSWER**

```
bool equalDigits(int x)
{
    if (x < 10)
        return true;
    else
        return x%10 == x/10%10
            && equalDigits(x/10);
}
```

4. Write a function **mset** that takes a map<int, int> by reference and changes all the values so they are equal to the keys. The function should return the number of values changed. The map size should not be changed. For example (25 points)

```
map<int, int> m;
m[9] = 5;
m[6] = 6;
m[4] = 0;
int i = mset(m); // your function
cout << m.size(); // 3
cout << i; // 2
cout << m[4]; // 4
cout << m[6]; // 6
cout << m[9]; // 9
```

**// ANSWER**

```
int mset(map<int, int>& m)
{
    int count = 0;
    map<int, int>::iterator p;
    for (p=m.begin(); p!=m.end(); ++p)
    {
        if (p->second != p->first)
            ++count;
        p->second = p->first;
    }
    return count;
}
```