QUEENS COLLEGE Department of Computer Science
CSCI 111 Midterm 2 Exam Spring 2016 05.04.16

Solutions

09.00am – 09.50am, Wednesday, May 04, 2016

Problem 1 *(points)* Write the best **title lines** for the functions that are called by the following main program. **Do not supply blocks for the functions.**

```
int main() {
   double x = 0.0, y = 1.1, z = 2.5;
   int array[5] = {3,1,4,1,5};
   string s = "Hello";

   z = average(x, y, z);                  // (a) sets z to average 1.2
   addStar(s);                            // (b) replaces s by "Hello*"
   cout << bigger(average(x,y,z), 7.5);   // (c) prints 7.5 because it is bigger
   cout << endl;
   printArray(array, 5);                  // (d) prints array: 3 1 4 1 5
   subtract(array[0], array, 5);          // (e) subtracts array[0] from other elements
   printArray(array, 5);                  //   output will now be 0 -2 1 -2 2
   return 0;
}
```

(a) Title line for **average**.

Answer:

```
double average(double a, double b, double c)
```

(b) Title line for **addStar**.

Answer:

```
void addStar(string &x)
```

(c) Title line for **bigger**.

Answer:

```
double bigger(double a, double b)
```

(d) Title line for **printArray**.

Answer:

```
void printArray(int a[], int cap)
```

(e) Title line for **subtract**.

Answer:

```
void subtract(int x, int y[], int cap)
```

Problem 2 *(points)* Consider the following C++ program.

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

int fun(int x, int &y) {
  if (x < 0) y = -x;
  if (x <= 0) return 0;
  return x % 10 + 2 * fun(x/100, y);
}

int main() {
    int c, x = 1, y = 5;
    if ((x % y) > (y % x)) cout << x;          // line (a)
    cout << endl;
    for(c = x; c < y; c++) cout << c;          // line (b)
    cout << endl;
    cout << fun(-2, y) << endl;                // line (c)
    cout << y << endl;                         // line (d)
    cout << fun(31459, y) << endl;             // line (e)
}
```

(a) What is the output at line (a)?
Answer:

1

(b) What is the output at line (b)?
Answer:

1234

(c) What is the output at line (c)?
Answer:

0

(d) What is the output at line (d)?
Answer:

2

(e) What is the output at line (e)?
Answer:

29

Problem 3 *(points)* Write a function called *subtractFirst* that subtracts the value of the first element from every element in an array.

For example, a program that uses the function *subtractFirst* follows.

```
int main() {
   int array[6] = {3,1,4,1,5,9};
   subtractFirst(array, 6);
   for (int i = 0; i < 6; i++)
      cout << array[i] << " ";   //   Output will be 0 -2 1 -2 2 6
   return 0;
}
```

Answer:

```
void subtractFirst(int array[], int c) {
   for (int i = c - 1; i >= 0; i--)
      array[i] -= array[0];
}
```

Problem 4 *(points)* Write a function called *cutAfter7* that cuts a positive integer parameter after the first digit 7 that it contains. Parameters that are not positive should be returned without any change.

For example, a program that uses the function *cutAfter7* follows.

```
int main() {
   cout << cutAfter7(765) << endl;     // prints 7
   cout << cutAfter7(765765) << endl;  // prints 7
   cout << cutAfter7(666) << endl;     // prints 666
   cout << cutAfter7(107) << endl;     // prints 107
   cout << cutAfter7(107007) << endl;  // prints 107
   return 0;
}
```

Answer:

```
int cutAfter7(int x) {
   if (x <= 0) return x;
   int y = cutAfter7(x/10);
   if ((y % 10) == 7) return y;
   return x;
}
```

QUEENS COLLEGE Department of Computer Science
CSCI 111 Midterm 2 Exam Spring 2016 05.04.16

Solutions

02.45pm – 03.35pm, Wednesday, May 04, 2016

Problem 1 *(points)* Write the best **title lines** for the functions that are called by the following main program. **Do not supply blocks for the functions.**

```
int main() {
   double z = 2.5;
   int array[5] = {3,1,4,1,5};
   string s = "Hello";

   z = average(array, 5);                 // (a) sets z to average 2.8
   addTwice(s,"**");                      // (b) replaces s by "Hello**Hello**"
   cout << sum(average(array, 5), 1.2);   // (c) 4.0 the sum of 1.2 and the average
   cout << endl;
   cout << someArray(array, 3);           // (d) prints 3 elements: 3 1 4
   count(array[1], array, 5);             // (e) print count of copies of array[1] in array
   return 0;
}
```

(a) Title line for **average**.

Answer:

```
double average(int a[], int cap)
```

(b) Title line for **addTwice**.

Answer:

```
void addTwice(string &x, string y)
```

(c) Title line for **sum**.

Answer:

```
double sum(double a, double b)
```

(d) Title line for **someArray**.

Answer:

```
string someArray(int a[], int cap)
```

(e) Title line for **count**.

Answer:

```
void count(int x, int y[], int cap)
```

Problem 2 *(points)* Consider the following C++ program.

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

int xy(int x, string &y) {
  if (x < 0) y = "error";
  else y = "ok";
  if (x <= 0) return 5;
  return x % 10 + 10 * xy(x/100, y);
}

int main() {
    int c = 4, x = 1;
    string y;
    if ((x % c) == (c % x)) cout << c;        // line (a)
    cout << endl;
    for(c = 5; c > x; c--) cout << c;         // line (b)
    cout << endl;
    cout << xy(-2, y) << endl;               // line (c)
    cout << y << endl;                        // line (d)
    cout << xy(31459, y) << endl;            // line (e)
}
```

(a) What is the output at line (a)?
Answer:

(b) What is the output at line (b)?
Answer:

```
5432
```

(c) What is the output at line (c)?
Answer:

```
5
```

(d) What is the output at line (d)?
Answer:

```
error
```

(e) What is the output at line (e)?
Answer:

```
5349
```

Problem 3 *(points)* Write a function called *subtractAverage* that subtracts the average value of an array from every element in an array.

For example, a program that uses the function *subtractAverage* follows.

```
int main() {
   double array[6] = {3,1,4,1,5};   // has average 2.8
   subtractAverage(array, 5);
   for (int i = 0; i < 5; i++)
      cout << array[i] << " ";   //   Output will be 0.2 -1.8 1.2 -1.8 2.2
   return 0;
}
```

Answer:

```
void subtractAverage(double array[], int c) {
   double total = 0;
   for (int i = 0; i < c; i++) total += array[i];
   double average = total / c;
   for (int i = 0; i < c; i++)
      array[i] -= average;
}
```

Problem 4 *(points)* Write a function called *cutBefore7* that cuts a positive integer parameter before the first digit 7 that it contains. Parameters that are not positive should be returned without any change.

For example, a program that uses the function *cutBefore7* follows.

```
int main() {
   cout << cutBefore7(667) << endl;     // prints 66
   cout << cutBefore7(677) << endl;     // prints 6
   cout << cutBefore7(666) << endl;     // prints 666
   cout << cutBefore7(766) << endl;     // prints 0
   cout << cutBefore7(567567) << endl;  // prints 56
   return 0;
}
```

Answer:

```
int cutBefore7(int x) {
   if (x <= 0) return x;
   int y = cutBefore7(x/10);
   if ((x % 10) == 7 || (y < x/10)) return y;
   return x;
}
```