QUEENS COLLEGE Department of Computer Science
CSCI 111 Final Exam Spring 2017 05.22.17

Solutions

08.30am – 10.30am, Monday, May 22, 2017

Problem 1 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() {
   int i = 123, arr1[3] = {1, 2, 3} , arr2[2][2] = {{1, 0}, {2, 4}};
   double d1 = 1.23, d2 = 12.3;
   printLine(arr2, 2, 2);             // (a) prints: 1 0 2 4
   printFancy(arr1, 3);               // (b) prints: 1 * 2 ** 3 ***
   cout << doNothing (i, (int) d1);   // (c) prints: This is a useless function
   switchValues(d1, d2);    // (d) switches the values: now, d1 = 12.3 and d2 = 1.23
   cout << goodDayWishes();           // (e) prints: Have a good day
   return 0;
}
```

(a) Title line for **printLine**.
Answer:

```
void printLine(int x[][2], int r, int c)
```

(b) Title line for **printFancy**.
Answer:

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

(c) Title line for **doNothing**.
Answer:

```
string doNothing(int a, int b)
```

(d) Title line for **switchValues**.
Answer:

```
void switchValues(double &x, double &y)
```

(e) Title line for **goodDayWishes**.
Answer:

```
string goodDayWishes()
```

Problem 2 Write blocks of code to perform the functions used in the following main program. Your blocks must match the given title lines. Each block should be a short function of only a few lines.

```
int main() {
   int a[5] = {3, 1, 4, 1, 5}, b[5] = {2, 7, 1, 8, 2};
   string s = "Final", t = "Exam";
 // (a) Prints the array.
   printArray(a, 5);                  // output:  3 1 4 1 5
 // (b) Finds index of max entry.
   cout << maxIndex(a, 5) << endl;    // output:  4
 // (c) Swaps array entries
   swapArrays(a, b, 5);
   printArray(a, 5);                  // output: 2 7 1 8 2
 // (d) find piece of t starting at: a (assume a is present).
   cout << cutFrom(t, "a") << endl;   // output:  am
 // (e) determine whether s or t has more characters
   if (hasMore(s,t)) cout << "s is longer\n";
   return 0;
}
```

Answer:

(a)

```
void printArray(int x[], int c) {
    for (int i = 0; i < c; i++) cout << x[i] << " ";
    cout << endl;
}
```

(b)

```
int maxIndex(int x[], int c) {
    int ans = 0;
    for (int i = 1; i < c; i++)
      if (x[i] > x[ans]) ans = i;
    return ans;
}
```

(c)

```
void swapArrays(int x[], int y[], int c) {
   for (int i = 0; i < c; i++) {
      int temp = x[i];
      x[i] = y[i];
      y[i] = temp;
   }
}
```

(d)

```
string cutFrom(string x, string target) {
    return x.substr(x.find(target));
}
```

(e)

```
bool hasMore(string x, string y) {
   return x.length() > y.length();
}
```

Problem 3 Consider the following C++ program.

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

int func1(double &d, string s) {
   s = "Final Exam";
   d = 13.14 - 3.14;
   cout << "s" << endl;
   return 13 + 1;
}

int func2 (int &a, int &b, int c) {
   a = b + c;
   b = 1;
   return c;
}

int main() {
   double piDoubled = 3.14 + 3.14;
   string str = " CSCI ";
   func1 (piDoubled, str);                  // line (a)
   cout << func1(piDoubled , str) << endl;  // line (b)
   cout << piDoubled << piDoubled  << endl; // line (c)
   int x = 1 , y = 11 ;
   cout << 2 * (func2(x, y, x)) << endl;    // line (d)
   cout << x << y << endl;                  // line (e)
   return 0;
}
```

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

Answer:

```
s
```

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

Answer:

```
s
14
```

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

Answer:

```
1010
```

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

Answer:

```
2
```

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

```
121
```

Answer:

Problem 4 Write a function called *triPrint* that uses the entries of an array of characters to print a triangle. The first row of the triangle has the first entry, the second row the first two entries and so on. Your solution should use no more than 6 lines of code.)

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

```
int main() {
   char x[7] = {'c', 's', 'c', 'i', '1', '1', '1'};
   triPrint(x, 7);
   return 0;
}
```

The ouput from this program would be:

```
c
cs
csc
csci
csci1
csci11
csci111
```

Answer:

```
void triPrint(char x[], int c) {
   for (int r = 1; r <= c; r++) {
      for (int c = 0; c < r; c++) cout << x[c];
      cout << endl;
   }
}
```

Problem 5 Write a function called *swapTwo* that has an integer parameter that is at least 10. It returns an integer obtained by swapping the first two digits in the input number. If an argument less than 10 is given your function can return any result of your choosing.

Your function need not use more than 2 instructions. Excessively complicated long solutions that use more than 6 lines of code may lose points.

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

```
int main() {
   cout << swapTwo(19683) << endl;          // prints 91683
   cout << swapTwo(10) << endl;             // prints 1
   cout << swapTwo(swapTwo(19683)) << endl; // prints 19683
   return 0;
}
```

Answer:

```
int swapTwo(int x) {
   if (x < 100) return x / 10 + 10 * (x % 10);
   return 10 * swapTwo(x/10) + x % 10;
}
```

Problem 6 Write a complete C++ program that does the following:

1. Asks the user to enter 2 integers, x and y. Both should be between 2 and 10 (inclusive), and if either is illegal then the program terminates.

2. Fills a table (as part of a 2d-array) with characters entered by the user. The table should have as many rows as x and as many columns as the double of y. The user should enter the characters separated by spaces.

3. Prints the characters in the last column in reverse order without spaces.

For example, the following represents one run of the program:

Enter 2 integers : 3 2

Enter 12 characters : a b c d e f g h i j k l

The characters in the last column (reversed): lhd

Answer:

```
#include <iostream>
using namespace std;
int main() {
   int x, y, r, c;
   char table[10][20];
   cout << "Enter 2 integers: ";
   cin >> x >> y;
   if (x < 2 || x > 10 || y < 2 || y > 10) return 0;
   cout << "Enter " << 2 * x * y << " characters: ";
   for (r = 0; r < x; r++)
     for (c = 0; c < 2 * y; c++)
        cin >> table[r][c];
   cout << "The characters in the last column (reversed): ";
   for (r = x - 1; r >= 0; r--)
      cout << table[r][2*y - 1];
   cout << endl;
   return 0;
}
```

QUEENS COLLEGE Department of Computer Science
CSCI 111 Final Exam Spring 2017 05.22.17

Solutions

01.45pm – 03.45pm, Monday, May 22, 2017

Problem 1 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() {
   int arr1[3] = {1, 2, 3} , arr2[2][2] = {{1, 0}, {2, 4}};
   string s1 = "Final", s2 = "Exam";
   cout << max(arr2, 2, 2);    // (a) prints: 4
   cout << endl;
   printMax(arr1, 3);          // (b) prints max, here: 3
   cout << firstOne(s1, s1);   // (c) returns the first, here: Final
   cout << endl;
   switchValues(s1, s2);       // (d) switches the values: now, s1 = "Exam" and s2 = "Final"
   goodDayWishes(arr1[1], arr2[1][1]);       // (e) prints: Have a good day
   return 0;
}
```

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

Answer:

```
int max(int x[][2], int r, double c)
```

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

Answer:

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

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

Answer:

```
string firstOne(string a, string b)
```

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

Answer:

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

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

Answer:

```
void goodDayWishes(int x, int y)
```

Problem 2 Write blocks of code to perform the functions used in the following main program. Your blocks must match the given title lines. Each block should be a short function of only a few lines.

```
int main() {
   int i = 12;
   int x[5] = {3, 1, 4, 1, 5};
// (a) Return the largest odd factor.
   cout << oddFactor(i) << endl;     // output: 3
// (b) Return the sum of even entries.
   cout << sumEven(x, 5) << endl;    // output: 4
// (c) last digit of i.
   cout << lastDigit(i) << endl;     // output: 2
// (d) Find the (last) index of the smallest entry.
   cout << findIndexMin(x, 5) << endl;  // output: 3
// (e) Is it upper case?
   if (isUpper('h')) cout << "Digit" << endl;    // No output here.
   return 0;
}
```

Answer:

(a)

```
int oddFactor(int x) {
    if (x % 2 != 0) return x;
    return oddFactor(x / 2);
}
```

(b)

```
int sumEven(int array[], int cap) {
   int ans = 0;
   for (int i = 0; i < cap; i++)
      if (array[i] % 2 == 0) ans += array[i];
   return ans;
}
```

(c)

```
int lastDigit(int x) {
   return x % 10;
}
```

(d)

```
int findIndexMin(int array[], int cap) {
   int ans = 0;
   for (int i = 0; i < cap; i++)
      if (array[i] <= array[ans]) ans = i;
   return ans;
}
```

(e)

```
bool isUpper(char x) {
   return 'A' <= x && x <= 'Z';
}
```

Problem 3 Consider the following C++ program. It is compiled to **a.out** and executed with the command **./a.out 007**.

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

int fun(int &x, int y, int &z) {
   y = x;
   x = z;
   z = y;
   cout << z;
   return x;
}

int main(int argc, char *argv[]) {
   int x = 3, y = 1, z = 4;
   fun(x, y , z); cout << endl;              // line (a)
   cout << x << y << z << endl;              // line (b)
   cout << fun(x, y , z) << endl;            // line (c)
   cout << argc << endl;                     // line (d)
   cout << argv[1] << endl;                  // line (e)
   return 0;
}
```

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

Answer:

```
3
```

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

Answer:

```
413
```

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

Answer:

```
43
```

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

Answer:

```
2
```

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

Answer:

```
007
```

Problem 4 Write a function called *sums* that replaces each entry in an array of integers by the sum of that entry and all earlier entries in the original input array. Your solution should use no more than 6 lines of code.)

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

```
int main() {
   int x[6] = {3, 1, 4, 1, 5, 9};
   sums(x, 6);
   for (int i = 0; i < 6; i++) cout << x[i] << " ";
   cout << endl;
   return 0;
}
```

The ouput from this program would be:

```
3 4 8 9 14 23
```

because, for example $3 + 1 + 4 + 1 + 5 = 14$ and $3 + 1 + 4 + 1 + 5 + 9 = 23$.

Answer:

```
void sums(int x[], int c) {
   if (c <= 1) return;
   sums(x, c - 1);
   x[c - 1] += x[c - 2];
}
```

Problem 5 Write a function called *bigGap* that has an integer parameter that is at least 10. It returns an integer that gives the biggest gap between adjacent digits in the input number. If an argument less than 10 is given your function can return any result of your choosing.

Your function need not use more than 6 instructions. Excessively complicated long solutions that use more than 12 lines of code may lose points.

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

```
int main() {
   cout << bigGap(19683) << endl;           // prints 8 found as the gap in 19
   cout << bigGap(38691) << endl;           // prints 8 found as the gap in 91
   return 0;
}
```

Answer:

```
int bigGap(int x) {
   int lastGap = (x / 10) % 10 - x % 10;
   if (lastGap < 0) lastGap = -lastGap;
   if (x < 100) return lastGap;
   int earlierGap = bigGap(x / 10);
   if (earlierGap > lastGap) return earlierGap;
   return lastGap;
}
```

Problem 6 Write a complete C++ program that does the following:

1. Asks the user to enter an integer x. It should be between 2 and 10 (inclusive), and if it is illegal then the program terminates.

2. Makes the user to enter x words (strings) of text, each of which should have at least 4 characters. Any word with fewer characters is replaced by the string ”Error”.

3. Prints the third character from each word, beginning with the last word and ending with the first.

For example, the following represents one run of the program:

Enter an integer : 3

Enter 3 words: Final Exam CSCI111

The third characters in reverse order: Can

Answer:

```
#include <iostream>
using namespace std;
int main() {
   int x, r;
   string words[10];
   cout << "Enter an integer: ";
   cin >> x;
   if (x < 2 || x > 10) return 0;
   cout << "Enter " << x << " words: ";
   for (r = 0; r < x; r++)
        cin >> words[r];
   cout << "The third characters in reverse order: ";
   for (r = x - 1; r >= 0; r--)
      cout << words[r][2];
   cout << endl;
   return 0;
}
```