STUDENT NUMBER:______________________

THE HONG KONG UNIVERSITY OF SCIENCE & TECHNOLOGY

COMP 102: Computer Fundamentals and Programming

Fall Semester, 1997

FINAL EXAMINATION

December 10, 19967

STUDENT NAME:	______________________
STUDENT NUMBER:	______________________
DEPARTMENT:	______________________
SECTION NUMBER:	______________________
LAB SECTION:	______________________

Instructions to Students

1. There are 15 problems and 17 pages. Check that you have all 17 pages.
2. Write your name, student no., department, section no., and lab no. on this page.
3. Write your student number on each of the following pages.
4. Answer all questions in the space provided. Rough work should be done only on the back pages and afterwards drawn a diagonal line through it to show that it is not part of your answer.
5. Leave all pages stapled together.
6. The examination period will last for **180 minutes**.
7. Stop writing immediately when the time is up.

For Grading Purposes Only

Problem 1	________	/ 8
Problem 2	________	/ 4
Problem 3	________	/ 4
Problem 4	________	/ 8
Problem 5	________	/ 8
Problem 6	________	/15
Problem 7	________	/ 6
Problem 8	________	/ 6
Problem 9	________	/ 5
Problem 10	________	/ 8
Problem 11	________	/ 6
Problem 12	________	/ 6
Problem 13	________	/ 6
Problem 14	________	/ 4
Problem 15	________	/ 6
TOTAL:	________	**/ 100**

1. (8 points) Write an int function that takes two integer arrays of the same size and returns their dot product. To compute the dot product, multiply the arrays position by position and sum the products. For example, with a[3] = {2,3,4} and b[3]={1,2,3}, the dot product is 2*1 + 3*2 + 4*3 = 20.

The function prototype is:

```
int dot_product(const int a[], const int b[], int size);
```

2. (4 points) What is the exact output of the following program?

```
#include <iostream.h>
int main()
{
    int i=3, j=5, k=7;
    for(int n=1; n<(i*j)%10; n++)
     {
        k = (n*k)%10;
        cout << k;
    }
  cout << endl;
  return 0;
}
```

3. (4 points) What is the content of the array A[] upon completion of the for loop in the following code segment?

```
int A[10] = {7, 2, 10, 5, 3, 6, 4, 8, 9, 1};

for (int count = 0; count < 9; count++)
  if (A[count] > A[count+1])
  {
     int temp = A[count];
     A[count] = A[count+1];
     A[count+1] = temp;
  }
```

	0	1	2	3	4	5	6	7	8	9
A										

4. (8 points) Given a 3x3 array A declared as

```
int A[3][3];
```

write a bool function that tests whether, for all *i*, the sum of the *i*th row equals the sum of the *i*th column. For example, the array b[3][3] = {{2, 4, 5}, {4, 3, 9}, {5, 9, 7}} has this property, but the array c[3][3] = {{2, 4, 6}, {4, 3, 9}, {5, 9, 7}} does not.

The function prototype is:

```
bool is_equal_sums(const int A[3][3]);
```

5. **(8 points) Write a function that is given an integer array and its size, and reverses the array values. For example, a[4] = {0,1,2,3} should become a[4] = {3,2,1,0). Assume that the function prototype is:**

```
void reverse_array(int a[], int size_a);
```

6. (15 points) Here is an example of a marks file.

```
92384758 9 10  10 83
98389058  10 9 10  72
92584757   8       10  8 83
92392658 8 10 10 82
92378258   9 9 10         83
```

The marks file has five columns, the first column contains student id numbers and the remaining four columns contain the marks for four exams of an unspecified course. There are at most 30 students in each marks file and the marks are all integers.

Write a program that queries the user for the name of the marks file, opens the marks file, reads the marks file into one two-dimensional array, outputs the average mark for each exam, and closes the marks file.

(Question 6 continues)

7. (6 points) You are given the following binary search function:

```
int binary_search(const int data[],//input array
                  int size_data, //size of data[]
                  int query_value //search value
                  ) //return index of query_value
                    //in data[] if found;
                    //otherwise, return -1
{
   int lower = 0, upper = size_data - 1;
   int middle;

   while (lower < upper)
   {
      middle = (lower + upper)/2;
      if (query_value <= data[middle])
          upper = middle;
      else lower = middle + 1;
   }
   if (query_value == data[lower])
       return lower;
   else return -1;
}
```

If

```
int array d[10] = {2,3,5,7,11,13,17,19,23,29},
```

then what are the values returned by the following calls and how many times is the body of the while loop executed in each case:

a. binary_search(d,10,23);

b. binary_search(d,10,1);

8. (6 points) What is the output of the following recursive function when called with an argument of 3?

```
void recursive(int i)
{
  if (i < 6)
   {
    recursive(i+1);
    cout << i;
   }
}
```

a) 456

b) 345

c) 543

d) 654

e) None of the above

9. (5 points) What is the output given by the following code?

```
struct Date {
  int y, m, d;
};
 struct Time {
  int hour, min;
 };
 struct Appointment {
  Date date;
  Time from, to;
 };

Appointment appnts[] = {
 { {1997, 12, 10}, {12, 30}, {15, 30} },
 { {1997, 12, 24}, {18, 00}, {23, 59} },
 { {1998,  1,  1}, { 0, 0},  {1, 0} }
 };

 cout << appnts[0].date.d << " "
      << appnts[0].from.hour << ":"
      << appnts[0].from.min << endl;

 cout << appnts[1].date.d << "/"
      << appnts[1].date.m << "/"
      << appnts[1].date.y << endl;
```

10. (8 points) Write an int function *vowel* that returns the number of vowels in a given string. Assume that a, e, i, o, u, and y are vowels.

The function prototype is either:

```
int vowel(char a[]);
```

if you use character arrays (C strings),

or

```
int vowel(string a);
```

if you prefer to use the string class.

11. (6 points) You are given the following bool function:

```
bool what(int a[], int first, int last)
{
    int size = last - first + 1;
    if (size == 1)
        return true;
    else if (size == 2)
        return a[first] <= a[last];
    else
        return what(a, first, first + size/2) &&
               what(a, first + size/2, last);
}
```

You are also given the following declaration and initialization:

```
int b[6] = {10, 11, 19, 35, 21, 17};
```

What values are returned by the following two calls of what:

a. `what(b,0,3);`

b. `what(b,2,4);`

c. What does the function what() achieve? Describe its purpose in at most two sentences.

12. (6 points) The following program is supposed to read a sequence of positive integers terminated by -1 and print out the rangelength of the sequence, where the rangelength is defined as the difference between the maximum and minimum values. For example, the input sequence 34 15 4 26 17 9 48 21 5 -1 should give the output

```
rangelength = 44
```

Assume further that the input integers are not greater than 32767 and the sequence ends with a -1. Identify any bug(s) in the program and correct the mistake(s).

```
#include <iostream.h>

int main()
{
   int i, max, min;
   cin >> i;
   while (i != -1)
   {
        if (i > max) max = i;
        if (i < min) min = i;
        cin >> i;
    }
   cout << "rangelength = " << max - min << endl;
   return 0;
}
```

13. (6 points) Given an int array, its size, and a query integer, the following function is supposed to search for the query value in the array. If the query value is found, its index is returned; otherwise, -1 is returned. Is the function correctly implemented? If not, explain the problem and fix it.

```
// Searches an unordered array of integers
int search(int data[], // input: array
           int size,   // input: array size
           int query   // input: value to find
           )           // return: index, if the query value
                       // is found;
                       // otherwise, -1

{
     int index = 0;    // initialize index to 0

     while ((index < size) && (data[index] != query))
          index++;

     if (data[index] != query)
          return -1;
     else
          return index;
} // search
```

14. (4 points) You are given the following program:

```
#include <iostream.h>
int main()
{
    int  count;
    char sym[5];

    for (count = 0; count < 5; count++)
        cin >> sym[count];
    for (count = 4; count > 0; count--)
        cout << sym[count];
    cout << endl;
    return 0;
}
```

Suppose that you run this program and enter the following input (<ENTER> **is the Enter key and** ␣ **indicates a space):**

A␣␣C␣<ENTER>

<ENTER>

$5<ENTER>

#<ENTER>

What is the output?

15. (6 points) Show the exact output of the following program.

```
#include <iostream.h>

int main()
{
    int array[4][4];
    int index1, index2;
    for (index1 = 0; index1 < 4; index1++)
        for (index2 = 0; index2 < 4; index2++)
            array[index2][index1] = index2;
    for (index1 = 0; index1 < 4; index1++)
    {
      for (index2 = 0; index2 < 4; index2++)
            cout << index1 << array[index1][index2]
                 << index2 << " " ;
        cout << endl;
    }
    return 0;
}
```