CS1010E Programming Methodology
Tutorial 6: Arrays

> That is what learning is. You suddenly understand something you've understood all your life, but in a new way.
>
> ~ *Doris Lessing*

To students:

Welcome back! We hope that you have made good use of the one-week recess to revise on the topics covered so far. Now we are moving on to a new phase, one that focuses more on **logical thinking** and **problem solving**. Practice makes perfect!

I. Manual Tracing

A bunch of data structures (to be introduced in CS1020 and CS2010) are implemented using arrays to store and expand their data. Thus arrays are massively used in programming, as they provide a very fast and easy way to store and access large amount of data.

1. Trace the following program manually and write down the output. Thereafter, copy the program from PDF file and run it on sunfire to verify the result.

```
#include <stdio.h>
#define LENGTH 5

void print_array(int arr[], int size);
void process(int arr[], int size);

int main(void) {

  int numbers[] = {2, 1, 3, 0, 4};
  process(numbers, LENGTH);
  print_array(numbers, LENGTH);

  return 0;
}

void process(int arr[], int size) {
  int i;
  for (i = 0; i < size; i++) {
    arr[i] = arr[arr[i]];
  }
}
void print_array(int arr[], int size) {
  int i;
```

```
  for (i = 0; i < size; i++) {
    printf("%d ", arr[i]);
  }
  printf("\n");
}
```

2. [CG1101 AY2011/2012 Semester 1 Exam, Q1e]

   Trace the following program manually and write down the output.

```
#include <stdio.h>

int func(int arr[], int x, int y);

int main(void) {
  int i, arr[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

  arr[0] = func(arr, 0, 9);
  arr[1] = func(arr, 1, 2);

  for(i = 0; i < 10; i++) {
    printf("%d ", arr[i]);
  }
  printf("\n");

  return 0;
}

int func(int arr[], int x, int y) {
  int temp = arr[y];
  arr[y] = arr[x];
  return temp;
}
```

3. Trace the following program manually and write down the output. What do you learn from this exercise?

```
#include <stdio.h>
void pass_element(int num);
void change_elements(int mylist[]);
void print(int mylist[], int arraySize);
int main(void) {
  int list[] = {11, 22, 33, 44, 55};

  printf("Original array:\n");
  print(list, 5);

  pass_element(list[0]);
  printf("After passing one element:\n");
  print(list, 5);

  change_elements(list);
  printf("After changing individual elements:\n");
  print(list, 5);

  return 0;
}

void pass_element(int num) {
  num = 1234;
}

void change_elements(int mylist[]) {
  mylist[0] = 66;
}

void print(int mylist[], int arraySize) {
  int i;
  for (i = 0; i < arraySize; i++) {
     printf("%d ", mylist[i]);
  }
  printf("\n");
}
```

4. [CS1010 AY2011/2012 Semester 1 Midterm Test, Q16]

   Consider the following function.

```
int do_something(int arr[], int n) {
  int i, j;
  for (i = 0; i < n; i++) {
    for (j = 0; j < n; j++) {
      if (i != j && arr[i] == arr[j]) {
        return 1;
      }
    }
  }
  return 0;
}
```

(a) Assuming **arr** is an array of integers and **n** is the number of elements in the array, describe the purpose of the above function? Keep your answer concise.

(b) Write an improved version of the function to make it more efficient.

II. Hands-on Session

You may download the programs of Q5 and Q6 as follows.

```
cp ~cs1010/tutorial/week7_q*.c .
```

5. This is a true story. We once received an email from a lecturer about the following program which, when run, gives an infinite loop.

   Give a possible explanation. What is the moral of the story?

```
#include <stdio.h>

int main(void) {

  double arr[] = { 1.1, 2.2, 3.3, 4.4 };
  int i;

  for (i = 0; i <= 4; i++) {
    printf("%d ", i);
    arr[i] = 0;
  }

  printf("%.2f\n", arr[0]);
  return 0;
}
```

6. Spot the syntax errors in the following program and fix them.

```
#include <stdio.h>
```

```
double sum_array(double arr, int size);

int main(void) {

  double[6] prices, total;
  prices = {10.2, 5.3, 4.4, 6.8, 7.7, 9.5};

  sum_array(prices[6], 6);

  printf("Total = %.2f\n", total);

  return 0;
}

double sum_array(double arr, int size) {

  int i;
  double sum = 0.0;

  for (i = 0; i <= size; i++) {
    sum += prices[i];
  }

  return sum;
}
```

7. **[Problem Set 3 Exercise #03] Is Array Sorted?**

8. **[Problem Set 3 Exercise #05] Positive First**

9. **[Problem Set 3 Exercise #09] Most Appeared Digit**