ES112 Summer 2003-2004 Final

1 Second Largest Element *(25 points)*

Your task in this question is to write a **function** that finds the **index** of the second largest element of a given array of `ints`. You may not change the order of elements in the array (so you are not allowed to sort it first), you may not make a copy of the array, you may not assume a maximum size for the array, and you may not allocate any memory using `malloc` or anything similar. The prototype of the function is:

```
int index_of_second_largest(int *a, int n);
```

Here, `n` is the number of elements in the array, and `a` is a pointer to the start of the array. The function should return the index of the second largest element in the array.

2 Calculate Change *(25 points)*

Change ("bozuk para") in America comes in (mainly) four types: Quarters (25 cents), dimes (10 cents), nickels (5 cents), and pennies (1 cent). Your task in this question is to write a **function** that calculates and prints how to give a given amount of change with the least number of coins. The prototype of the function is:

```
void print_change(int cents);
```

For instance, if we call the function as `print_change(37)` the output should be:

```
37 cents:
Quarters: 1
Dimes: 1
Pennies: 2
```

If the function is called with an argument less than zero or greater than 100, it should print nothing.

3 Is It a Text File? *(25 points)*

Write a C **function** that takes a filename as an argument, and if that file exists, determines whether it is a text file or not. The code should decide that it is a text file if more than 95% of the characters in the file are **printable characters**. If the file is **not** a text file, it should return zero, and if it is a text file, it should return 1. The prototype is given below:

```
int is_textfile(const char *filename);
```

You may find the following library function useful:

```
int isprint(int c);

Checks for any printable character including space. Returns nonzero if
printable, zero otherwise.
```

4 What Is Going On? *(25 points)*

a. What does the below code print?
b. What does the function `hoppa` do, in general?

```
#include <stdio.h>

void hoppa(int *a, int *b)
{
  *a ^= *b;
  *b ^= *a;
  *a ^= *b;
}

int main(void)
{
  int x=3, y=7;

  hoppa(&x, &y);
  printf("x=%d y=%d\n", x, y);
  return 0;
}
```