CS 515 Take-Home Worksheet #1

Name: Nick Snyder UNH User ID: nrs1053 Total: /100 pts

Lecture Section: 01 / 02 (circle one)

Part 1: Functions & Pointers [50 pts]

Consider the following program fragment for the questions below:

```
const int MAX_NAME_LEN = 81;
const int NUM_RESULTS = 5;
int read_dimensions(int *rows, int *cols );
void print_results(char names[][MAX_NAME_LEN], double values[][NUM_RESULTS]);
double det(double *a[], int rows, int cols);

int main()
{
      double *matrix[3];
      char names[10][MAX_NAME_LEN] = {"Virginia", "Philip", "Rhoda"};
      double times[NUM_RESULTS][10] = {{0}};
      double averages[10][NUM_RESULTS] = {{0}};
      int r, c, s, i = 3;
      int * p, *q;
      p = &i;

      for (i = 0; i < 3; i++)
      {
          matrix[i] = new double[3];
      }
```

(a) For each of the following, indicate which statements would be valid at this point in the main function by marking the appropriate box. "Valid" means a statement would not likely cause syntax or runtime errors. If a statement is invalid, provide a reason why it is invalid.

❑ Valid	❑ Invalid	`*p = 7;`
❑ Valid	❑ Invalid	`print_results(names, times);`
❑ Valid	❑ Invalid	`averages[0][0] = det(matrix, 3, *p);`
❑ Valid	❑ Invalid	`c = *q;`
❑ Valid	❑ Invalid	`&q = 5;`
❑ Valid	❑ Invalid	`print_results(names, averages);`
❑ Valid	❑ Invalid	`det(names, r, c);`
❑ Valid	❑ Invalid	`s = read_dimensions(&r, &c);`
❑ Valid	❑ Invalid	`p = 15;`

(b) Excluding print_results, name the function that expects two parameters to be passed by reference.

read_dimensions()

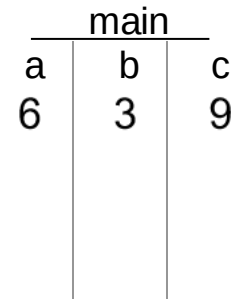
(c) Excluding print_results, name the function that expects two parameters to be passed by value.

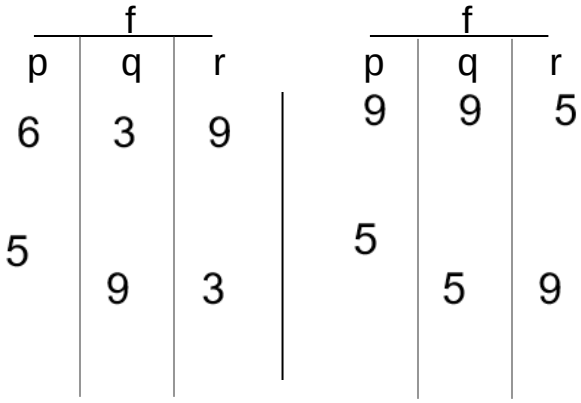
det()

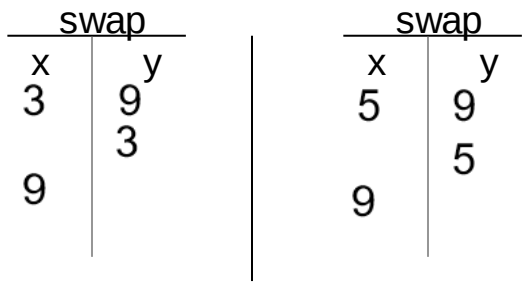
Part 2: Tracing Execution [50 pts]

What is the output from this program (spacing doesn't matter—for the output portion, just show what would be printed out)? ***You must show your work to get credit.***

```
#include <iostream>
using namespace std;
void f(int p, int * q, int * r);
void swap(int *x, int *y);

int main(void)
{
    int a = 6 , b = 3, c = 9;

    cout<<"M1: "<< a <<" "<< b <<" "<< c <<endl;
    f(a , &b, &c) ;
    cout<<"M2: "<< a <<" "<< b <<" "<< c <<endl;
    f(c , &b, &a) ;
    cout<<"M3: "<< a <<" "<< b <<" "<< c <<endl;
    return 0;
}


void f(int p, int * q, int * r)
{
    cout << "F1: "<< p << " " << *q
         << " " << *r << endl;
    p = 5;
    if (*q < *r)
        swap(q, r);
    else
        swap(&p, q);
    cout << "F2: "<< p << " " << *q
         << " " << *r << endl;
    return;
}

void swap(int *x, int *y)
{
    int val = *y;
    *y = *x;
    *x = val;
    cout << "SWAP: " << *x << " "
         << *y << endl;
}
```







Output:

M1: 6 3 9

F1: 6 3 9

SWAP: 9 3

F2: 5 9 3

M2: 6 9 3

F1: 6 9 3

SWAP: 9 5

F2: 5 5 3

M3: 6 5 6