PASS MOCK EXAM – FOR PRACTICE ONLY

Course: SYSC 2006 C Facilitator: Madeline Harlow
Dates and locations of mock exam take-up: Monday March 4, 6 pm in SP 100

IMPORTANT:

It is **most beneficial** to you to write this mock midterm **UNDER EXAM CONDITIONS**. This means:

- Complete the midterm in 1.5 hour(s).
- Work on your own.
- Keep your notes and textbook closed.
- Attempt every question.

After the time limit, go back over your work with a different colour or on a separate piece of paper and try to do the questions you are unsure of. Record your ideas in the margins to remind yourself of what you were thinking when you take it up at PASS.

The purpose of this mock exam is to give you practice answering questions in a timed setting and to help you to gauge which aspects of the course content you know well and which are in need of further development and review. Use this mock exam as a *learning tool* in preparing for the actual exam.

Please note:

- Come to the PASS session with your mock exam complete. There, you can work with other students to review your work.
- Often, there is not enough time to review the entire exam in the PASS session. Decide which questions you most want to review – the facilitator may ask students to vote on which questions they want to discuss.
- Facilitators do not bring copies of the mock exam to the session. Please print out and complete the exam before you attend.
- Facilitators do not produce or distribute an answer key for mock exams. Facilitators help students to work together to compare and assess the answers they have. If you are not able to attend the PASS session, you can work alone or with others in the class.

Question 1 [18 marks] Fundamental Programming Syntax

a) [2 marks] What will be outputted from the following code fragments?

i) [1 marks]

```
int pointer = 5, x = 7;
int * y = &x;
*y = pointer;
pointer = *y + x;
x = *y + pointer;
printf("%d, %d, %d.\n", pointer, x, *y);
```

ii) [1 marks]

```
char c1 = 'a', c2 = c1+3;
char *p = &c1;
*p += 1;
*p++;
(*p)++;
printf("%c, %c, %c.\n", c1, c2, *p);
```

b) [2 marks] Make the necessary changes to properly initialize the elements.

```
struct question1 {int a; char s[80];};

char * str = "Hello World";
struct question1 *p = (struct question1*)malloc(sizeof(struct question1));
_____ = 5;
_____ = str;
free(p);
```

c) [14 marks] Given that student information is saved in a file passed into a program by command line arguments, write the program to open and read the contents of the file into a struct array of correct size (the struct is given to you). A sample file is shown below, where the first line containing the number of entries, followed by the entries themselves. Error checking is required.

```
6
Chuck Testa          100738593   54
Ron Burgendy         100345765   78
Rick Roll            100938576   32
Barack Obama         100293543   87
Tony Stark           100098432   99
Felix Baumgartner    100128100   28
```

```
typedef struct{
   char first[80];
   chart last[80];
   int num;
   int grade;
}STUDENT;
```

Question 2 [22 marks] Functions and Stack Frame Diagrams

a) [7 marks] Convert the following code into Pass-By-Reference and draw the stack frame diagrams at the given points for the code you have written (for extra practise, draw the stack frame diagram for the given code as well).

```
typedef struct{
      int num;
      int dnum;
}fraction;

fraction makeFraction (int n, int d){
      fraction fr = {n, d};
      //(i) Draw diagram at this point
      return fr;
}
int main (void){
      int x = 4, int y = 9;
      fraction z = makeFraction (y, x);
      int w = z.num/z.dnum;
      //(ii) Draw diagram at this point
      return 0;
}
```

b) [5 marks] What will be the output of the following code fragment? Draw stack frame diagram(s) to show your work.

```
int main (void) {
      int array[] = {5, 2, -9, 6, 3};
      int temp[] = {0, 0, 0, 0, 0};
      int *list = temp;
      for(int i = 0; i<5; i++){
            temp[i] = array[i] + 1;
            *list = array[i] + 1;
            list++;
      }
      printf("array\ttemp\tlist\n");
      for(int i = 0; i<5; i++){
            printf("%d\t%d\t%d\n", array[i], temp[i], *list++);
      }
}
```

c) [10 marks] For the following function prototypes which would be used in the same program as Question 1 C, write the implementation, show how to test the function and draw the stack frame diagram just before the return.

```
/*
 *@param   list  An array of STUDENT's
 *         size  Size of array
 *@return  the location of the highest grade found in the array
 */
STUDENT* highestGrade (STUDENT* list, int size);

/*
 *@param   student    variable of type STUDENT
 */
void printStudent (STUDENT student);
```