# CURTIN UNIVERSITY
# Department of Computing

## Test 1 – Semester 2, 2017

**SUBJECT**: COMP1000 - UNIX AND C PROGRAMMING

**TIME ALLOWED**:
90 minutes

**GENERAL INSTRUCTIONS**:

This is a CLOSED BOOK test. Put away all books, notes, etc. Turn off and put away all electronic devices, including calculators and watches. (Make sure your phone is switched off.)

You must write in blue or black pen – pencil will not be accepted.

There are FIVE (5) questions. Write your answers on the question paper. Use the back of the pages if needed (especially for working out).

You have 90 minutes to complete the test. There are 90 marks available.

**ATTEMPT ALL QUESTIONS**

**Markers Use only – do not write in these boxes**

| Q1 | Q 2 | Q 3 | Q 4 | Q 5 | Total /90 |
|---|---|---|---|---|---|
| | | | | | |

## Question 1 (10 marks)

Clearly and fully explain the meaning of the following:

(a) `r(&t);`

[2 marks]

(b) `float *a =*b;`

[2 marks]

(c) `*x = *y.`

[2 marks]

(d) `float* g = (*h)(i);`

[2 marks]

(e) `#define Q(p, r) ((int)(*(p) + (r)))`

[2 mark]

## Question 2 (20 marks)

Draw complete, labelled diagrams of memory and/or pointer relationships for the below code segment. Stack memory should start at 1000. The datatype `int` occupies 4 bytes of memory. Additionally, show the values of all non-pointer variables.

Hint: You have been given an example of two different methods to do this. Either will suffice.

```
int avocado, broccoli;
int *cabbage, *dill;
int **eggplant, **fennel, **garlic;

eggplant = &cabbage;
fennel = &dill;
garlic = fennel;
*eggplant = &broccoli;
**eggplant = 9;
*fennel = &avocado;
**garlic = 5 - broccoli;

fennel = eggplant;
**fennel += **eggplant;
```

**Write your answer neatly on the next page**

**Write your answer neatly on this page**

## Question 3 (15 marks)

In a GIF viewer program, there are three .c files: ui.c, loader.c and render.c. As per convention, each .c file includes its own header file. In addition, ui.c includes the other two header files, and render.c includes loader.c's header file and uses the math library.

Produce a suitable `makefile` for this project, in the form emphasised in the lectures and tutorials. The executable file should be named `gifviewer`.

## Question 4 (10 marks)

Show the output from each of the following C programs:

(a)
```
#include <stdio.h>

double silly(int);

int main(void)
{
   int i;
   for(i = 1; i < 10; i += 2)
   {
      printf("%lf\n", silly(i));
   }
   return 0;
}

double silly(int t)
{
    static int x = 1;
    static int y = 0;
    y += x * t;
    x = -x;
    return (double)y / 2.0;
}
```

[6 marks]

(b)
```
#include <stdio.h>

#define DIV(a,b) ((float)(a / b))

int main(void)
{
        int r = 3, s = 9;
        printf("%f\n", DIV(7, 3));
        printf("%f\n", DIV(s, r));
        printf("%f\n", DIV(4 + 3, 2 + 1));
        return 0;
}
```

[4 marks]

## Question 5 (35 marks)

Write a C function (not a whole program) called calcPrices that will calculate final prices in a fishing/camping store.

Your colleagues have written several functions for applying discounts to a single product. They'd like your function to be able to call any one of their functions for each product. Your function must also ensure that the price remains positive. It should disregard any discounts that would make the price zero or negative.

Your function must return nothing, and accept the following parameters:

- `nCategories` — The number of categories of products.
- `averageDiscount` — The average discount applied to all products across all categories (as a pointer, for export purposes).
- `discountCalc` — A pointer to one of your colleagues' discounting functions. All of their functions takes a pointer to a float, representing a price. The function updates (reduces) the value stored at that location, or leaves it alone. The function returns nothing.

Each product category contains a number of different types of products. Categories are identified by an integer index, between zero and nCategories minus one. Within each category, products are also identified by an integer index, starting at zero. Your function must step through each product in each category, setting the new price. Your function should also calculate the average discount.

You will need to use three more functions which are provided for you. ie you do not need to write these:

- To retrieve the number of different types of products in a category:
```
int getNProducts(int category);
```
- To retrieve the original product price:
```
float getPrice(int category, int product);
```
- To set the new price:
```
void setPrice(int category, int product, float newPrice);
```

Use the following blank pages to answer this question. Ensure your code conforms to the standards emphasised in the lectures and tutorials.

Hint: write pseudo code first! Also, space out your answer as much as possible. You can amend your answer to insert extra code, but you must make it neat and legible. There are two blank pages for working out.

**Write your answer neatly on the next page**

**Write your answer neatly on this page**

**Use this page for rough draft**