



University College Dublin
An Coláiste Ollscoile, Baile Átha Cliath

SEMESTER 2 EXAMINATION – 2010/2011

Computer Science for Engineers

COMP 10060

Prof. M. McGinnity
Prof. Joe Carthy
Prof. J. Murphy*

Time allowed: 2 hours

Instructions for candidates

Answer **Question 1** and **one** other Question.
Question 1 carries 60 marks; Questions 2 and 3 carry 40 marks.

This is a closed-book examination. No calculators allowed.
Loose Rough Work sheets are not to be distributed or used.

READ EACH QUESTION CAREFULLY.

Instructions for Invigilators

This is a closed-book examination. No calculators allowed.
Loose Rough Work sheets are not to be distributed or used.

Question 1 (COMPULSORY) [60 marks]

Answer all parts (a) – (t). Each part carries 3 marks.

- (a) What is the screen output of the following fragment of C code (no explanation required):

```
double x=42.58;
printf("value is %7.3f\n",x);
```

- (b) Suppose that x, y, z and w are all of type int. If the value of w is 9, x is 20, the value of y is 3, and the value of z is 2, what is the value of w in the expression?

```
w /= x / (y + y * z)
```

- (c) What is the screen output of the following fragment of C code (no explanation required):

```
float number = 435.73810, med ;
med = ((int) (100*number))/100.0 ;
printf("number is %7.4f\n",med);
```

- (d) What is the screen output of the following fragment of C code (no explanation required):

```
int i=0,j=1;
for (i=6;i>2;i--){
    j*=i;
    printf("j is %d\n",j);
}
```

- (e) What is the screen output of the following fragment of C code (no explanation required):

```
double x=3.14159;
printf("value is %.4f",-x);
```

- (f) What is the screen output of the following fragment of C code (no explanation required):

```
int i=12;
while (i>6){
    i -= 2;
    printf("i is %d\n",i);
}
```

- (g) What is the screen output of the following fragment of C code (no explanation required):

```
#include <stdio.h>
int f1(int a, int b, int c){
    return ((a+b)*c);
}
int f2(int a, int b){
    int c = f1(a, b, 3);
    return c+1;
}
int main(void)
{
    printf("result is %d\n", f1(-1,f2(2,4),2));
    return 0;
}
```

- (h) What is the screen output of the following fragment of C code (no explanation required)?

```
int x = -3;
int y = 2;
int* p = &y;
*p = (*p)*x + (*p)*y;
printf("x is %d and y is %d\n", x, y);
```

- (i) What is the screen output of the following fragment of C code (*no explanation required*)?

```
char str[]="abcdefghijklmn";
char wanted[]="cdhjln";
int i, j;
for (j=0; wanted[j]!='\0'; j++){
    for (i=0; str[i]!='\0'; i++){
        if (str[i]==wanted[j]){
            str[i]='W';
            break;
        }
    }
}
printf("string=%s\n", str);
```

- (j) What is the screen output of the following fragment of C code *(no explanation required)*?

```
struct student {
    char name[30];
    char surname[30];
    int id;
    int age;
    char hobby[30];
};

int main(void)
{

    struct student me;
    me.id = 69854632;
    printf("output is %d\n", me.id);

    return 0;
}
```

- (k) What is the screen output of the following fragment of C code *(no explanation required)*?

```
int i;
int somearray[5] = {5,7,6,9,8};
for (i = 0; i <= 4; i++){
    printf("Element number %d is %d\n", i,
somearray[i]);
}
```

- (l) What is the screen output of the following fragment of C code *(no explanation required)*?

```
int count(int A[], int size, int target) {
    int n = 0, i;
    for (i=0; i<size; i++) {if (A[i]==target)
{n++;}}
    return n;
}

int main(void)
{
int name[]={4,6,5,3,2,5,6,7,8,9,1,3,4,5};
int result=0;
result = count(name,15,5);
printf("output is %d\n", result);

    return 0;
}
```

- (m) What is the screen output of the following fragment of C code (no explanation required):

```
int i;
for (i=1;i<15;i++){
    switch(i)
    {
        case 3: printf("3");
        case 4: printf("4\n");
        break;
        case 11: printf("11");
        case 12: printf("12\n");
        default: printf("default\n");
        i+=i;
    }
}
```

- (n) What is the screen output of the following fragment of C code (no explanation required):

```
int a=2, b=0, c=-2;
if (a&&b||c){
    printf("Condition is true\n");
} else {
    printf("Condition is false\n");
}
```

- (o) What is the screen output of the following fragment of C code (no explanation required):

```
int j=0, y;
y = ! j;
printf("y is %d\n",y);
```

- (p) What is the screen output of the following fragment of C code (no explanation required):

```
int nstars = 5, stars;
while (nstars >= 1) {
    stars = 1;
    while (stars <= nstars) {
        printf("*");
        stars++;
    }
    printf("\n");
    nstars--;
}
```

- (q) What is the screen output of the following fragment of C code (no explanation required)?

```
int i=-5,j=-i;
if (j<=-10){
    printf("first\n");
} else if ((-i)>=(-j)){
    printf("second\n");
} else {
    printf("no match\n");
}
```

- (r) What is the screen output of the following fragment of C code, given that the initial values in datafile.txt are:

0.2 80
0.7 60
0.1 50

```
float prob, avg=0.0;
int quantity, num_values=0;
FILE *fp;
fp=fopen("datafile.txt", "r");
while(fscanf(fp, " %f %d", &prob, &quantity)==2){
/* if return value from this fscanf() is not 2, */
/* the end-of-file indicator has been reached */
avg = avg + (prob*quantity);
}
fclose(fp);
printf("average value is %.2f\n", avg);
```

- (s) What are the contents of datafile.txt after the execution of this fragment of code, given that the initial values in datafile.txt are:

200
-54

```
int var1, var2;
FILE *fp;
FILE *fp2;
fp=fopen("datafile.txt", "r");
fscanf(fp, "%d %d", &var1, &var2);
fclose(fp);
fp2=fopen("datafile.txt", "w");
fprintf(fp2, "%d\n%d", var2, var1);
fclose(fp2);
```

- (t) What is the screen output of the following fragment of C code (no explanation required)?

```
int count(int A[], int size, int target) {
    int n = 0, i;
    for (i=0; i<size; i++) {if (A[i]==target)
{n++;}}
    return n;
}

int main(void)
{
int name[]={4,6,5,3,2,5,6,7,8,9,1,3,4,5};
int result=0;
result = count(name,12,5);
printf("output is %d\n", result);

    return 0;
}
```

Question 2 [40 marks]

Answer all parts (a) – (c).

- (a) The following fragment of C code adds up the integers from 1 to 10 inclusive using a **while** loop:

```
int sum = 0, i = 1;
while (i <= 10) {
    sum += i;
    i++;
}
```

Re-write this code fragment using a **for** loop instead of the **while** loop.

[10 Marks]

- (b) Consider the following C program:

```
#include <stdio.h>
/* DEFINITION OF FUNCTION "zerofinder" GOES HERE */
void main(void) {
    int i, array2[8]={1,-1,-1,0,1,0,-1,1};
    i = zerofinder(array2,8);
    if (i == -1) printf("\nno zero value in array2\n");
    else printf("\nfirst 0 element array2 has index %d\n", i);
}
```

Write down the definition of function **zerofinder()** which returns the index of the first 0 value, or -1 if no 0 value is found, so that the output of the above program is:

first 0 element array2 has index 3

[15 Marks]

- (c) Consider the following C program:

```
#include "stdio.h"
void main(void){
    char message[80]="I love C programming";
    int i=0, count=0;
    while (message[i]!='\0'){
        if ((message[i]=='c') || (message[i]=='C')){
            count++;
        }
        i++;
    }
    printf("\n%s" contains %d c's\n", message, count);
}
```

Re-write the above program so that all the code for determining the number of **c**'s in the string **message** is contained in a function called **c_counter()** that you should define, while the declaration and initialization of **message** and the output is still done from **main()**.

[15 Marks]

Question 3 [40 Marks]

Answer all parts (a) – (b).

(a) Consider the following C program:

```
#include "stdio.h"
void main(void){
    char message[80]="hello there";
    int count=0;
    while (message[count]!='\0'){
        count++; /* increase by 1 as long as NULL not reached */
    }
    printf("\"%s\" has length %d\n", message, count);
}
```

(i) What is the screen output of this program (*no explanation required*)?

(ii) **Re-write** the above program so that all the code for determining the length of the string **message** is contained in a function called **stringlength()** that you should define.

[20 Marks]

(b) Write a program that uses a function **convert()** to determine the equivalent number of hours, minutes, and seconds for a given time in seconds. For example, 3,661 seconds is equivalent to 1 hour, 1 minute, and 1 second. Use the following function prototype:

```
void convert(int time, int *phrs, int *pmins, int *psecs);
```

Remember to use good programming standards, define the problem statement, show the design of the program, add comments to the code, and follow good formatting practices to lay out the code. Your answer should include the algorithm design and the code.

[20 Marks]