

UNIVERSITY COLLEGE DUBLIN

NATIONAL UNIVERSITY OF IRELAND, DUBLIN

An Colaiste Ollscoile Baile Atha Cliath

Ollscoil na hEireann, Baile Atha Cliath

SUMMER EXAMINATIONS 2004

FIRST EXAMINATION IN ENGINEERING

Programme Codes: ENBDF0002, ENBDF0003,
ENBDF0004, ENBDF0005, ENBDF0008, ENBDF0011

COMP1604 COMPUTER SCIENCE

Prof. J. Kramer

Mr. G. O'Hare

Dr. L. Murphy*

Time allowed: 2 hours

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.

Question 1 (COMPULSORY) [60 marks]

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

- (a) True or False (*no explanation required*): in C, all declarations and statements **must** end with a “;”
- (b) True or False (*no explanation required*): in C, the programmer can decide which **memory location** is to be used to store a particular variable in their program.
- (c) True or False (*no explanation required*): “_3float” is a valid identifier in C.
- (d) If $x==7$, $y==3$, and $z==3$, what is the value of w in the expression $w = x / y * z$?
- (e) What is the screen output of the following fragment of C code (*no explanation required*):

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

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

```
int a=0, b=2, c=-1;
if ((a+c) || ((b/2)>0) ){
    printf("condition is true\n");
} else {
    printf("condition is false\n");
}
```

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

```
int i=99;
while (i>0){
    i = (i+1)/10;
    printf("i is %d\n",i);
}
```

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

```
int j = 3;
do {
    printf("j is %d\n", j);
    j--;
} while (j > 0);
```

- (i) True or False (*no explanation required*): if a C function which does not return a value contains a **return** statement, this is detected by the compiler as a syntax error.

[Question 1 continues]

Question 1 (continued)

- (j) Select the correct answer: The function prototype

```
void fn(int *x);
```

tells us that

- (j-1) **fn()** takes no arguments and returns a value of type **int**.
(j-2) **fn()** takes 1 argument of type **int** and does not return a value.
(j-3) **fn()** takes 1 argument of type pointer-to-**int** and does not return a value.

- (k) What is the screen output of the following C program (*no explanation required*):

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

- (l) Select the correct answer: If you want to multiply the first and third elements of an array **arr[]** together, you should use

- (l-1) **arr[0]*arr[2]**
(l-2) **arr[0]*[2]**
(l-3) **arr[1]*arr[3]**

- (m) True or False (*no explanation required*): If **i** and **j** are integer variables and **arrflts[]** is an array of **floats**, **arrflts[i*j]** is a valid expression in C.

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

```
int a=3, b=-4, *ptr=&a;
*ptr = b;
printf("a is %d, b is %d\n", a, b);
```

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

```
int x = 0;
int y = 2;
int* p = &y;
*p = y*y;
x = (*p) + 1;
p = &x;
printf("x is %d and y is %d\n", x, y);
```

[Question 1 continues]

Question 1 (continued)

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

```
double x[3] = {1.5, 2.2, 4.3};
printf("value is %.2f\n", *(x+1) );
```

- (q) True or False (*no explanation required*): to store the string "Engineering", we need an array of type **char** with a size of at least 11 characters.

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

```
int i;
char str[]="I hope to do well on this Exam.";
for (i=0; str[i+1]!='\0'; i++){
    if ((str[i]==' ') && (str[i+1] >= 'a') && (str[i+1] <= 'z')) {
        str[i+1] += 'A' - 'a';
    }
}
printf("str=%s\n", str);
```

- (s) Select the correct answer: in a data file, a **Sentinel signal** means

- (s-1) a special value (not part of the data) that tells the number of data "records" in the file.
- (s-2) any control values (not part of the data) contained in the file.
- (s-3) a value outside the range of actual data values indicating that the end of the data has been reached.

- (t) Given the following definition and declaration:

```
struct Employee {
    int number;
    char name[30];
    int age;
    char position[30];
};
struct Employee emp1;
```

Which of the following statements correctly assigns the value **47** to **emp1**'s age?

- (t-1) **emp1.age = 47;**
- (t-2) **emp1->age = 47;**
- (t-3) **emp1-age = 47;**

Question 2 [40 marks]

Answer all parts (a) – (c).

(a) The following fragment of C code adds up the integers from 1 to 5 inclusive using a **while** loop:

```
int sum = 0, i = 1;
while (i <= 5)
    sum = sum + i;
    i++;
}
```

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

(b) The following fragment of C code determines the type of fuel **f** using **if/else-if/else** statements:

```
char f;
/* suppose a value is now entered for f - code not shown */
if (f=='u'){
    printf("unleaded petrol\n");
} else if (f=='p'){
    printf("premium petrol\n");
} else if (f=='d'){
    printf("diesel\n");
} else printf("incorrect value entered\n");
```

Re-write this code fragment using a **switch** statement instead of the **if/else-if/else** statements.

(c) Consider the following C program:

```
#include <stdio.h>
/* DEFINITION OF FUNCTION "zerofinder" GOES HERE */
void main(void) {
    int i, array1[8]={1,-1,-1,0,1,0,-1,1};
    i = zerofinder(array1,8);
    if (i == -1) printf("\nno zero value in array1\n");
    else printf("\nfirst zero element of array1 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 zero element of array1 has index 3

Question 3 [40 marks]

Answer parts (a) and (b).

- (a) Consider the following C program:

```
#include "stdio.h"
void main(void){
    char message[80]="I like C programming";
    int i=0, count=0;
    while (message[i]!='\0'){
        if (message[i]== ' '){ /* test for blank space */
            count++;
        }
        i++;
    }
    printf("\'%s\' has %d spaces\n", message, count);
}
```

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

- (b) Consider the following C program:

```
#include <stdio.h>
void main(void)
{
    FILE *fptr1, *fptr2;
    int inp;
    fptr1 = fopen("input.dat", "r");
    fptr2 = fopen("output.dat", "w");
    while (fscanf(fptr1, "%d", &inp)==1){
        if (inp >= 0){
            fprintf(fptr2, "%d ", inp);
        }
    }
    fclose(fptr1);
    fclose(fptr2);
}
```

- (i) Suppose the file **input.dat** contains the following data:

```
0
-2
2
6
-1
-5
```

Complete this sentence:

After executing this program, the file **output.dat** contains _____.

- (ii) *Re-write* the above program so that it writes all values between -1 and 4 (inclusive) in the file **input.dat** to a new file called **output2.dat**

o O o