



EXAMINATIONS — 2007

MIDTERM TEST

COMP 206
PROGRAM AND
DATA STRUCTURES

Time Allowed: 90 minutes

Instructions: There are five questions:

- Question 1 is worth 18 marks
- Question 2 is worth 12 marks
- Question 3 is worth 16 marks
- Question 4 is worth 20 marks
- Question 5 is worth 24 marks

The exam will be marked out of 90.

You should attempt all the questions.

Write your answers in the exam paper and hand in all sheets.

Operator precedences are given in Appendix A.

There are spare pages at the end of the book for extra working, if required.

Student ID:

Question 1.

[18 marks]

(a) [12 marks]

In the box below give the output of the following program:

```
#include <stdio.h>

int main()
{
    int c = 4;
    int *p;
    char a[] = "COMP 206";
    float matrix[2][3] = {{1.0, 2.0, 4.0}, {8.0, 16.0, 32.0}};

    printf("a = %s\n", a);
    printf("*a + 3 = %c\n", *a + 3);
    printf("*(&c) = %d\n", *(&c));
    printf("matrix[1][1]= %f\n", matrix[1][1]);
    printf("*matrix[1] + 1 = %f\n", *matrix[1] + 1);
    printf("(*matrix)[0] = %f\n", (*matrix)[0]);

    return 0;
}
```

(Question 1 continued on next page)

Student ID:

(Question 1 continued)

(b) [6 marks] You want to write a program which will take the name of a file as an argument and write it to the terminal. Suppose that the file `art1.txt` has the English text of Article 1 of the Treaty of Waitangi. Then you would get this behaviour:

```
%gcc moreorless.c -o moreorless
%./moreorless art1.txt
```

```
The Chiefs of the Confederation of the United Tribes of New Zealand and the
separate and independent Chiefs who have not become members of the Confederation
cede to Her Majesty the Queen of England absolutely and without reservation all
the rights and powers of Sovereignty which the said Confederation or Individual
Chiefs respectively exercise or possess, or may be supposed to exercise or to
possess over their respective Territories as the sole sovereigns thereof.
```

You choose to implement the program by:

- opening a file;
- calling a function, `copyout`, to copy the file to the terminal;
- closing the file.

Complete the boxes in the following code:

```
#include <stdio.h>
```

```
/* prototype of copyout function */
```

```
{
    FILE *ifp;

    ifp = fopen(argv[1], "r"); /* open for reading */

    copyout(ifp);             /* copy the file out */

    fclose(ifp);             /* close the file */

    return 0;
}
```

(Question 1 continued on next page)

Student ID:

(Question 1 continued)

```
/* copyout definition */
```

Student ID:

Question 2.

[12 marks]

(a) [12 marks]

You want to write a C function `lift` which takes two integer variables and sets them both to be equal to the larger. Complete the details required in the boxes below. This output should be generated:

```
$gcc test1.c
$ ./a.out
x = 1
y = 0
x = 1
y = 1
```

```
#include <stdio.h>
```

```
/* lift prototype */
```

```
int main()
{
    int x = 0;
    int y = 1;
    printf("x = %d\ny = %d\n", x, y);
```

```
/* call to lift x and y */
```

```
printf("x = %d\ny = %d\n", x, y);
return 0;
}
```

(Question 2 continued on next page)

Student ID:

(Question 2 continued)

```
/* lift definition */
```

Student ID:

Question 3.

[16 marks]

A *shape* can be either a circle, a square or a rectangle. To define the size of a circle we need to know its radius; to define a square we need to know the length of the side; to define a rectangle we need to know its length and breadth.

The following C type definitions allow us to represent a shape:

```
typedef enum {circle, square, rectangle} Geometry;

typedef struct {
    float length;
    float breadth;
} LengthBreadth;

typedef union{
    float radius;
    float side;
    LengthBreadth sides;
} Size;

typedef struct {
    Geometry shape;
    Size size;
} Shape;
```

(Question 3 continued on next page)

Student ID:

(Question 3 continued)

(b) [8 marks] In the box below, define a function with prototype `float area(Shape s)`; which will return the area of a shape:

- the area of a circle is 3.14159 times the square of the radius,
- the area of a square is the square of the length of the side,
- the area of a rectangle is the length times the breadth.

```
float area(Shape s){
```

```
}
```

Student ID:

Question 4.

[20 marks]

(a) [6 marks]

In the box below state what you consider to be the three most significant **advantages** which C has when compared to Java.

(Question 4 continued on next page)

Student ID:

(Question 4 continued)

(b) [6 marks] In the box below state what you consider to be the three most significant **disadvantages** which C has when compared to Java.

Student ID:

(Question 4 continued)

(c) [8 marks] The C compiler works in several stages:

- preprocessing,
- compilation,
- assembly, and
- linking

In the box below state what happens at the preprocessing stage, and what preprocessor directives are used for.

Student ID:

Question 5.

[24 marks]

The following type definition and function prototypes are part of a system for recording data about students. The information about the students in a class is stored in a linked list. Privacy legislation prohibits us from keeping any more information than the students' ID numbers.

```
typedef struct student {
    int id;
    struct student *next;
} Student;
typedef Student* Class;

void printlist(Class); /* print/traverse a linked list */
int count(Class);     /* count the number of elements in the class */
void enroll(Class, int id); /* student with ID id joins the class */
void drop(Class, int id); /* student with ID id leaves the class */
```

(a) [8 marks] Write the C code to implement the function count to count the number of students in the class.

(Question 5 continued on next page)

Student ID:

(Question 5 continued)

(b) [8 marks] Write C code to implement function `drop` so that the student with a given ID is removed from the class.



Student ID:

(Question 5 continued)

(c) [8 marks] Write C code to implement function `enroll` so that the student with a given ID is added to from the class.

A large empty rectangular box with a thin black border, intended for the student to write their C code implementation of the `enroll` function.

Student ID:

A Operator Precedence

- 1 `()`, `->`, `[]`, `.`
- 2 `~`, `++`, `--`, `+` (unary), `-` (unary), `*`(unary), `&`(unary) , `sizeof`, (type)
- 3 `*`, `/`, `%` (arithmetic binary)
- 4 `+`, `-` (arithmetic binary)
- 5 `<<`, `>>`
- 6 `<`, `<=`, `>`, `>=`
- 7 `==`, `!=`
- 8 `&`
- 9 `^`
- 10 `|`
- 11 `&&`
- 12 `||`
- 13 `?:`
- 14 `=`, `+=`, `-=`, ... (assignment)
- 15 `,`

Student ID:

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.
