Student ID: .....

TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2012
MID-YEAR

NWEN 241

SYSTEMS PROGRAMMING

Time allowed: THREE HOURS

Instructions: The examination contains 7 questions. You must answer ALL questions

The exam consists of 180 marks in total, distributed to each of the questions as follows:

Question 1 Python fundamentals	[36 marks]
Question 2 Writing and reading Python programs	[36 marks]
Question 3 C Fundamentals	[29 marks]
Question 4 Arrays and Pointers	[26 marks]
Question 5 Dynamic Data Structures	[20 marks]
Question 6 Bitwise Operators	[16 marks]
Question 7 File Handling	[17 marks]

No calculators are allowed.

Paper foreign to English language dictionaries are allowed.

No electronic dictionaries are allowed.

Question 1 Python fundamentals **[36 marks]**

a) [5 marks] Some of following strings are legal Python identifiers and some are not. For each one state if the name is legal and if not, explain why:

i) Lambda

ii) 4squared

iii) days_of_week

iv) square-root

v) daysOfWeek

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.

b) [16 marks] For each of the following Python keywords, briefly explain what it does and write a short piece of code to illustrate its use.

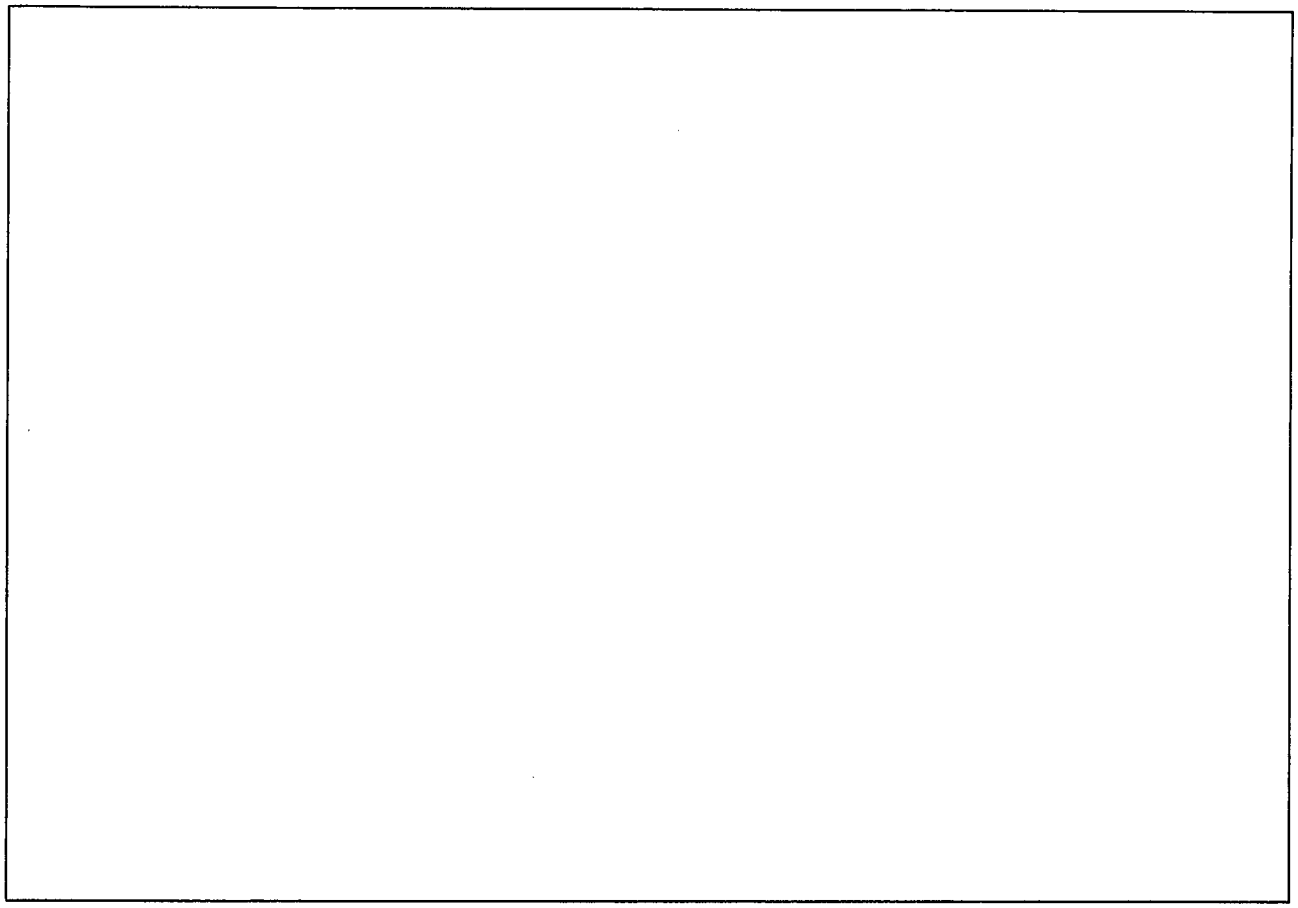
i) `try`

ii) `return`

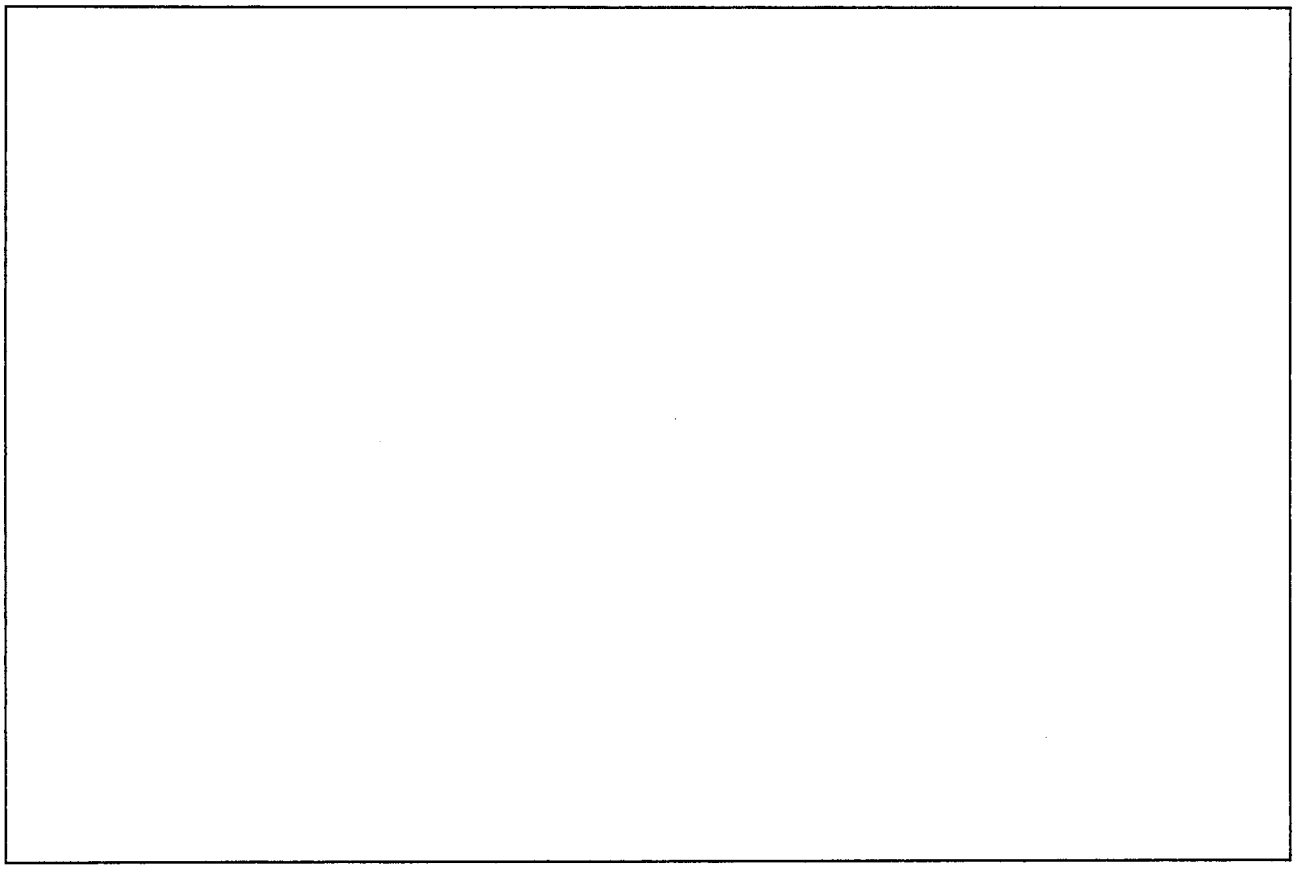
SPARE PAGE FOR EXTRA ANSWERS

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

iii) in



iv) import



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.

c) [15 marks] The greatest common divisor (GCD) of two integers, *a* and *b,* is the largest number that divides both of them with no remainder.

You have a systems programming task that requires extensive computation of the GCD of pairs of integers and you need to program this as efficiently as possible. You could choose to implement this in Python or in C.

Describe how you would test these two implementations to see which one would be a better choice. At a minimum, your description should address:

- use of system resources
- choice of test values
- the number of iterations of any tests

Question 2 Writing and reading Python programs **[36 marks]**

a) [10 marks] Write a Python program that reads two non negative integers as parameters on the command line and prints them out. For example, it might be called by:

```
$ python3 testprog.py 2 1
```

Your program should check the parameters and issue appropriate error messages and return values.

b) [10 marks] Modify your program to prompt the user for input if no command line arguments were given. You may choose to show just the changes, if you prefer.

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.

c) [16 marks] Consider the following python program which retrieves data from a website and analyses it. You are to add 10 suitable comments to the code explaining the section following.

```
#!/usr/bin/env python3

import datetime
import urllib.request

urlbase = 'http://list.waikato.ac.nz/pipermail/nznog/'
months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', 'October', 'November', 'December']

# Comment (1)
#
#
#
#
thisyear = datetime.datetime.now().year

# Comment (2)
#
#
#
#
years = range(1998, thisyear + 1)

def extract(text, sub1, sub2):
# Comment (3)
#
#
#
#
    return text.split(sub1, 1)[-1].split(sub2, 1)[0]

for year in years:
    for month in months:

# Comment (4)
#
#
#
#
        listurl = urlbase + str(year) + '-' + month + '/subject.html'
        try:
            fp = urllib.request.urlopen(listurl)
            mybytes = fp.read()
            fp.close()
        except:
            break

# Comment (5)
#
#
#
#
        encoding = extract(str(mybytes).lower(), 'charset=', '"')
        if encoding == None:
            print("Encoding type not found!")
            break

# Comment (6)
#
#
#
#
        lines = mybytes.decode(encoding).split('\n')
```

```
        totals = dict()
        total_subjects = 0
        total_postings = 0


        print (month, year, '\n***************')

# Comment (7)
#
#
#
#
        for line in lines:
            try:
                if '<LI>' in line:

# Comment (8)
#
#
#
#
                    posting = line.split('>')
                    url = posting[1].split('"')[1]
                    subject = posting[2].lstrip()

                    if subject not in totals:
                        totals[subject] = 1
                        total_subjects += 1
                    else:
                        totals[subject] += 1

                    total_postings += 1
            except:
                break

# Comment (9)
#
#
#
#
        for subject in sorted(totals, key=totals.get, reverse=True):
            print('{0:60} {1:10}'.format(subject, totals[subject]))

# Comment (10)
#
#
#
#
        print('\nPostings: {0:}, Subjects: {1:}\n'.format(total_postings,total_subjects))
```

Additional information

Data returned from the fp.read() statement in the code is a sequence of bytes that are encoded in a particular character set. You can assume that the HTML data retrieved will contain lines like the following:

```
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">

<LI><A HREF="019115.html">[nznog] APNIC 34 Conference - Call for Papers
</A><A NAME="19115"> </A>

<LI><A HREF="019073.html">[nznog] Beeping rack at Sky tower
</A><A NAME="19073"> </A>

<LI><A HREF="019074.html">[nznog] Beeping rack at Sky tower
</A><A NAME="19074"> </A>
```

Question 3 C Fundamentals **[29 marks]**

(a) [4 Marks] Explain the four steps of compilation for C programs.

(b) [4 Marks] State two or more important differences between C and Java.

(c) [4 Marks] Discuss the advantages and disadvantages of iteration versus recursion in C.

(d) [5 Marks] Assume the following malloc is successful:

```
int *ptr = malloc(20 * sizeof(int)); /* successful request */
```

Describe the possible outcomes of the following statement and discuss why a temporary variable `tmp` is used:

```
int *tmp = realloc(ptr, 200 * sizeof(int));
```

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.

(e) [4 Marks] Consider the following code:

```
int i;
float f;
...          /* f is assigned a value */
i = round_me(f);
```

`round_me()` rounds a `float` number to an integer. Give the function prototype/declaration of `round_me()`, and implement the function. You must not use any built-in functions.

(f) [4 Marks] Explain why the following code would not compile.

```
int *p1 = malloc(128);
extern i = 0;
int main(void)
{ char *p2 = malloc(128);
  ...
}
```

(g) [4 Marks] Consider the following `struct` type:

```
typedef struct {
  int i;
  float f;
} int_float;
```

Give a sensible definition of the function `multiply()`, which will "multiply" two `int-float` variables by multiplying the respective elements. For example, `multiply()` might be used in the following code.

```
int_float a, b, c;
...
c = multiply(a, b);
```

Question 4 Arrays and Pointers **[26 marks]**

(a) [6 Marks] Consider the following declaration.

```
char m[4][6] = {"01234", "56789", "abcde", "fghij", "klmno"};
```

Give the outputs of the following `printf` statements.

```
printf("%c", **m);
```

```
printf("%c", *(*m+3));
```

```
printf("%c", *(*(m+1)+3));
```

```
printf("%c", *(m[1]+2));
```

```
printf("%c", (*(m+4))[4]);
```

```
printf("%c", *(&m[0][0]+9));
```

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.

(b) [8 Marks] Give a declaration for the variable `p` in each of the following cases.

`p` is a pointer to an `int`.

`p` is a pointer to an array of 5 `int` elements.

`p` is a function that takes no arguments and returns a pointer to `int`.

`p` is an array of 10 pointers to functions that take no arguments and return an `int`.

`p` is a pointer to a pointer to a constant `char`.

`p` is a constant pointer to a `char`.

`p` is an array of `n` pointers to functions that take an `int` argument and return a pointer to functions that take a `char` argument and return a pointer to an array of `m` `int` elements.

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.

(c) [2 Marks] Consider the following declaration in the `main` function:

```
int *p;
```

Implement a standalone function that could be called to make `p` point nowhere.

(d) [10 Marks] Consider the following code using a function called `string_copy`:

```
char *s = "this is a string";
char *p = string_copy(s);
```

Write a definition of `string_copy` so that it copies its string argument to a new memory block and returns the base address of the memory block. You may use `string.h`.

Question 5 Dynamic Data Structures **[20 marks]**

(a) [4 marks] Consider the following code:

```
typedef struct {
  int i;
  char c;
  int ii;
} int_char_int;

typedef struct {
  int i;
  char c;
  char cc;
  int ii;
} int_char_char_int;
```

Suppose that you work on a 32-bit machine.

What is the value of `sizeof(int_char_int)`?

What is the value of `sizeof(int_char_char_int)`?

Suppose `a` and `p` are defined as follows:

```
int_char_int a;
int_char_int *p = &a;
```

If `p` currently contains the value `m`, what is the value of `p` after the statement `p++;`?

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.

(b) In this question, you need to implement functions that create a singly-linked list using iteration and recursion. You need to use the following type definitions, macro definitions and function prototypes to implement your functions:

```
#define Node_Size sizeof(Node)

typedef struct node
{ char data;
  struct node *next;
} Node;

typedef Node *ptrNode;

ptrNode createlisti(char *); /* iteration */
ptrNode createlistr(char *); /* recursion */
```

i. [8 Marks] Implement the function `createlisti()` using iteration. The function will create a list with one character per node from a string, and return a pointer to the head of the resulting list.

ii. [8 Marks] Implement the function `createlistr()` using recursion. The function will create a list with one character per node from a string, and return a pointer to the head of the resulting list.

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.

Question 6 Bitwise Operators **[16 marks]**

(a) [8 Marks] Suppose you are working on a 32-bit machine. Write a program that uses a mask and bitwise operators to print each bit of an integer. The program should get the user to type in an integer, and then should print the integer in the following format with spaces between blocks of 8 bits and a new line character at the end.

```
01001000 01101101 00001111 00010111
```

(b) [8 Marks] Write a definition of the function `bitwise_swap` that uses only bitwise assignment operators to swap the values of two strings (assume the two strings are of equal size).

Question 7 File Handling **[17 marks]**

For this question you need to write two functions which will write and read a singly-linked list to / from a binary file. The singly-linked list is constructed of nodes of the following `Node` type.

```
typedef struct node
{
  char data;
  struct node *next;
} Node;
```

Assume the function prototypes of `fwrite` and `fread` are as follows:

```
int fwrite(void *, int, int, FILE *);
int fread(void *, int, int, FILE *);
```

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.

(a) [8 Marks] Give an implementation of function `writelisttofile()` which uses `fwrite()` to write each of the nodes as a block of data to the file `list.dat`. `writelisttofile` is passed a pointer to the first node of the list. You need to declare the function with a proper function prototype and include an error message if the file cannot be opened.

(b) [9 Marks] Give an implementation of function `readlistfromfile()`, which uses `fread()` to read each of the nodes (a block of data) from the file `list.dat` and prints them on screen. For example, suppose the `data` value of the first node was `t` (character) and `next` was `bb902068` (hexadecimal), and the second node had `h` and `bb902070`. The output should look like this:

```
t bb902068
h bb902070
...
```

You need to declare the function with a proper function prototype and include an error message if the file cannot be opened.

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.

* * * * * * * * * * * * * * * * * * * *