Student ID: ....

TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI

VICTORIA
UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2013
TRIMESTER ONE

NWEN 241

SYSTEMS PROGRAMMING

Time allowed: THREE HOURS

Instructions: Closed book.

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

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

Question 1 Python Fundamentals	[30 marks]
Question 2 Writing and Reading Python Programs	[30 marks]
Question 3 C General Questions	[22 marks]
Question 4 Arrays and Pointers	[38 marks]
Question 5 Data Structures	[32 marks]
Question 6 Bitwise Operators	[16 marks]
Question 7 File Handling	[12 marks]

No calculators are allowed.

Paper foreign to English language dictionaries are allowed.

No electronic dictionaries are allowed.

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.

Student ID: ....

Question 1 Python Fundamentals [30 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.

Student ID: ....

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) `break`

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.

iii) in

iv) global

continued

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) [9 marks] You have a systems programming task that requires extensive string comparison and manipulation and you need to program this as efficiently as possible. You decide to implement this in Python and in C and compare the programs.

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

Student ID: ....

Question 2 Writing and Reading Python Programs **[30 marks]**

a) [8 marks] Write a Python program that reads two 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) [8 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) [14 marks] Consider the following python program which retrieves data from a website and analyses it. You are to write SEVEN(7) suitable comments explaining sections of the code under the box (numbered #1 to #7).

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

#
# (1)
#
#

import datetime
import urllib.request


#
# (2)
#
#

thisyear = datetime.datetime.now().year
years = range(1998, thisyear + 1)
months = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', 'October', 'November', 'December']
urlbase = 'http://list.waikato.ac.nz/pipermail/nznog/'

#
# (3)
#
#

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

for year in years:
    for month in months:
        listurl = urlbase + str(year) + '-' + month + '/subject.html'

#
# (4)
#
#

        try:
            fp = urllib.request.urlopen(listurl)
            mybytes = fp.read()
            fp.close()
        except:
            break

#
# (5)
#
#


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

        totals = dict()
        total_subjects = 0
        total_postings = 0

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

```
#
# (6)
#
#
```

```
        lines = mybytes.decode(encoding).split('\n')
        for line in lines:
            try:
                if '<LI>' in line:
                    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
```

```
#
# (7)
#
#
```

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

        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">&nbsp;</A>

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

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

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.

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.

Question 3 C General Questions **[22 marks]**

(a) [4 Marks] C provides static and automatic storage classes. Explain how a variable can be declared to be static or automatic, and explain the difference in behaviour between the two classes.

(b) [2 Marks] State the advantages of using `union` over `struct`.

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

(d) [4 Marks] C provides `malloc` and `free` for memory management. Explain what these functions do.

(e) [4 Marks] Briefly explain the difference between `malloc` and `calloc`. Discuss when `malloc` should be used in preference to `calloc` and vice-a-versa.

(f) [4 Marks] Suppose you are working on a 32-bit machine, where `sizeof(int)` is FOUR bytes, `sizeof(char)` is ONE byte and `sizeof(int *)` is FOUR bytes. Consider the following code:

```
#define mPchar char *
typedef char *tPchar;

mPchar ma, mb;
tPchar ta, tb;
```

Give the outputs of the following `printf` statements.

```
printf("%d ", sizeof(ma));
```

```
printf("%d ", sizeof(ta));
```

```
printf("%d ", sizeof(mb));
```

```
printf("%d ", sizeof(tb));
```

Student ID: ....

Question 4 Arrays and Pointers **[38 marks]**

(a) [14 Marks] Consider the following code.

```
int m[4][4] = {{1,3,5,7},{11,33,55,77},{2,4,6,8},{22,44,66,88}};
int (*parr)[4] = m;

int a[3] = {11, 22, 33};
int *pa = a;
```

Give the outputs of the following `printf` statements.

```
printf("%d", *a);
```

```
printf("%d", *(a+2));
```

```
printf("%d", *pa);
```

```
printf("%d", pa[1]);
```

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

```
printf("%d", *(*m+2));
printf("%d", *(*(m+1)+1));
```

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

```
printf("%d", (*(m+2))[3]);
```

```
printf("%d", (*(parr+3))[2]);
```

(b) [10 Marks] Give a declaration for the variable `p` in each of the following cases.

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

`p` is a array of 5 pointers to `char`.

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

`p` is a pointer to an array of 10 `int` elements.

p is a constant pointer to a constant char.

p is a pointer to an array of 5 pointers to function that takes no arguments and returns an int.

p is a function that takes no arguments and returns a pointer to function that takes an int argument and returns a pointer to an array of 10 int elements.

(c) [6 Marks] Define a function with prototype int string_length(char *); which returns the number of characters in the string.

(d) [8 Marks] Define a function with prototype `void string_reverse(char *);` which reverses the characters in a string. For example, the output of the following code

```
int main(void)
{ char str[] = "ABCDEFG";
  string_reverse(str);
  printf("%s \n", str);
  return 0;
}
```

should look like this:

```
GFEDCBA
```

You may use the string-handling function `strlen`, which returns the number of characters in string `s`, not counting the terminating NULL character.

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.

Question 5 Data Structures **[32 marks]**

(a) A person's height can be expressed in either metric or Imperial units.

- If we express a height in metric units we use metres and we give a floating point number, e.g. 1.75 m.
- If we express a height in Imperial units we use feet and inches and we give two integers, e.g. 5 foot 8 in.

Th following C type definitions allow us to represent a height:

```
typedef enum {metric, imperial} Scale;

typedef struct {
  int feet;
  int inches;
} FeetAndInches;

typedef union{
  float metres;
  FeetAndInches feetandinches;
} Value;

typedef struct {
  Scale scale;
  Value reading;
} Height;
```

i. [8 Marks] Define a function with prototype `void printheight(Height h);` which will print out a height, stating whether it is a height in metres or feet and inches.

ii. [8 marks] It is useful to be able to convert a height from feet and inches to metres. Define a function with prototype `void tometres(Height *h);`

Suppose that `h` is a height, which may be in either metres or feet and inches. After `tometres(&h)` has been called, `h` should be expressed in metres. There are 12 inches in a foot, and one inch is exactly 0.0254 metres.

(b) The following type definitions, macro definition and function prototypes are part of a queue model, where we are using singly-linked lists to implement queues.

```
#define Node_Size sizeof(Node)

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

typedef char Data;

typedef struct queue
{ int cnt;          /* counts the number of nodes */
  ptrNode front;    /* points to the front node */
  ptrNode rear;     /* points to the rear node */
} Queue;

void enqueue(Data, Queue *);
Data dequeue(Queue *);
```

Suppose the queue that we have implemented has a header node of type Queue and a list of linked nodes of type Node. The pointer front in the header node points to the front node in the list, while the pointer rear points to the rear node in the list. The header node also has a counter cnt, which counts the number of nodes in the list.

i. [8 Marks] Write C code to implement the function enqueue, which adds a new node to the rear of the list. The character passed to enqueue needs to be assigned to the variable data in the node. You can assume your requests for memory are always successful. You DO NOT need to consider the case when the queue is empty or full.

ii. [8 Marks] Write C code to implement the function `dequeue`, which deletes the front node from the list. The character stored in data needs to be returned. You DO NOT need to consider the case when the queue is empty, and you may assume the queue is never full.

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.

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.

Student ID: ....

Question 6 Bitwise Operators [16 marks]

In the following, we have defined a structure type named Student:

```
typedef struct student {
  int id;
  int age;
  char gender;
} Student;
```

(a) [8 Marks] Define a function with prototype `int pack(Student *);` which packs all the data members in a `Student` variable into an `int` variable and returns the value of this `int` variable. In this `int` variable, you must use **1 bit** to store `gender`, **7 bits** for `age` and **24 bits** for `id`.

(b) [8 Marks] Define a function with prototype `Student unpack(int);` which unpacks the integer that is retuned by the `pack` function in (a) into a `Student` variable. After unpacking, the values of this `Student` variable must exactly match the values of the `Student` variable used in (a). You may use `pow(x, y)`, which returns the value of `x` raised to the power of `y` to help create the masks that you need.

Question 7 File Handling **[12 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 *);
```

(a) [6 Marks] Define a function with prototype `void writelisttofile(Node *);` which uses `fwrite()` to write each of the nodes as a block of data to the file `list.dat`. You need include an error message if the file cannot be opened.

(b) [6 Marks] Define a function with prototype `void readlistfromfile(void);` 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 may assume that the file can always be successfully opened.

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.

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