

Student ID: .....



**EXAMINATIONS — 2015**  
**TRIMESTER ONE**

**NWEN 241**  
**SYSTEMS PROGRAMMING**

**Time allowed:** TWO HOURS

**CLOSED BOOK**

**Permitted materials:** No calculators are allowed.

No electronic dictionaries are allowed.

Paper foreign to English language dictionaries are allowed.

**Instructions:** The examination contains 3 questions. You must answer ALL questions

The exam consists of 120 marks in total, distributed across each of the questions as follows:

Question 1 C General Questions	[30 marks]
Question 2 Arrays, Pointers and File Handling	[60 marks]
Question 3 Bitwise Operators and Data Structures	[30 marks]

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 1 C General Questions**

**[30 marks]**

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

(b) [3 Marks] Explain the difference between Java array and C array.

(c) [3 Marks] Explain the difference between Java and C in terms of memory management.

- (d) [6 Marks] The program memory space has three types of sections. State these sections, how they are used, and whether they are used for compile time or run time memory allocation.

- (e) [6 Marks] Discuss the difference between `malloc` and `calloc`, and their advantages and disadvantages. Give an example where you may choose `calloc` over `malloc`.

(f) [4 Marks] Discuss the issues with the following macro definition:

```
#define SQ(x) x * x
```

(g) [4 Marks] Comment on the following statement – what is it about?

```
scanf ("%[^\\n] ", s);
```

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 2 Arrays, Pointers and File Handling****[60 marks]**

(a) Implement function `arrayAdd` that sets each element in an array `arr1` to the sum of the corresponding elements in two other arrays `arr2` and `arr3`. That is, if `arr2` has the values 1, 2, 3 and 4, and `arr3` has the values 4, 3, 2 and 1, the function assigns `arr1` the values 5, 5, 5 and 5. You may assume the three arrays have the same size `arr_size`.

i. [5 Marks] Use array notation to implement this function.

ii. [5 Marks] Use pointer notation to implement this function.

(b) Consider the following code.

```
int m[4][4]={ {0,1,2,3}, {4,5,6,7}, {8,9,10,11}, {12,13,14,15} };  
int (*p)[4]=m;
```

i. [7 Marks] Give the outputs of the following printf statements.

```
printf("%d", m[0][0]);
```

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

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

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

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

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

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



Student ID: .....

- ii. [8 Marks] Assume you are working on a 32-bit machine where the sizes of an `int` and a `void *` are both 4 bytes. Also assume the address of the first byte of `m[0][0]` is 0.

`p++`; What is the value of `p`?

What is the value of `&m`?

What is the value of `m+2`?

What is the value of `&m[2]`?

What is the value of `*(m+2)+2`?

What is the value of `m[3]`?

What is the value of `m[3]+1`?

What is the value of `&m[3]+1`?

(c) [7 Marks] Give a declaration for the variable `p` in each of the following cases.

`p` is a pointer to an element of a string.

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

`p` is a pointer to a string of 8 characters.

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

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

`p` is an array of 8 pointers to a function that takes no arguments and returns an `int`.

`p` is an array of 5 pointers to a function that takes two arguments: a pointer to `int` and a pointer to a pointer to `int` and returns a pointer to a function that takes an `int` argument and returns a pointer to an array of 5 `float` elements.

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: .....

- (d) [6 Marks] Write a definition of the function `strSearch` that takes a character and a string as its two arguments. `strSearch` searches the string. If the character is found, it replaces the character with 'F'. For example, if the character is 'a' and the string is "apple", after the execution of `strSearch`, the string will be "Fpple".

- (e) [8 Marks] Write a command-line-arguments based program. The program will be called with two file names as its command line arguments. The program should read the text from the first file and write it to the second file. You **must use pointer notation** to implement this program.

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.

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

```
int main(void)
{
    char a[2][10] = {"black", "red"};
    swap(*a, *(a+1));
    printf(...);      /* to be completed */
    return 0;
}
```

Implement function swap, which swaps the values between the two rows of array a, and complete the printf statement so that the outputs of the above program should look like this:

red, black

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 Bitwise Operators and Data Structures****[30 marks]**

(a) [10 Marks] In the following, we have defined a structure type named ToBePacked:

```
#define char_node_size sizeof(ToBePacked)
typedef struct ToBePacked ToBePacked;
typedef ToBePacked *ptrToBePacked;

struct ToBePacked {
    char data;
    ptrToBePacked next;
};
```

Write a function with prototype `ptrToBePacked charList(char *)`, which creates a ToBePacked node for each character in a string and links the nodes in sequence, and returns a pointer to the first node of the resulting list.



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.

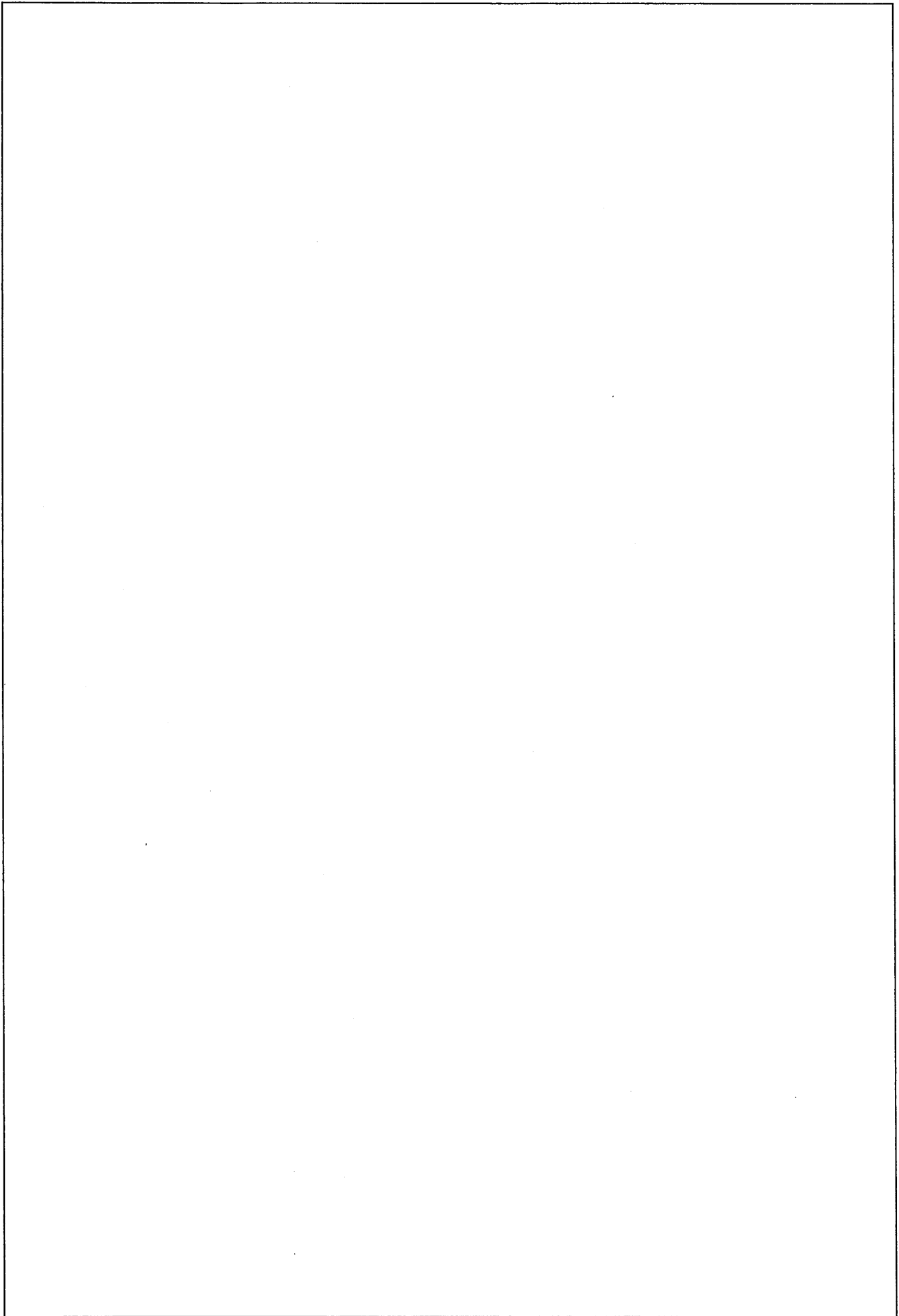
(b) [20 Marks] In the following, we have defined a structure type named Packed:

```
#define int_node_size sizeof(Packed)
typedef struct Packed Packed;
typedef Packed *ptrPacked;

struct Packed {
    int data;
    ptrPacked next;
};
```

Write a function with prototype `ptrPacked packCharacters(ptrToBePacked)`, which packs the chars from each of the ToBePacked nodes created in Question (a) into the int variable in a Packed node. Assuming the `sizeof(char)` is one byte and the `sizeof(int)` is four bytes, you need pack Four chars (from four ToBePacked nodes each) into the int in a Packed node. The function deallocates the memory space of the ToBePacked node after its char is packed. The function links the Packed nodes in sequence and returns a pointer to the first Packed node of the list.

Student ID: .....

A large, empty rectangular box with a thin black border, occupying the majority of the page. It is intended for a student to draw or write their answer.

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.

\*\*\*\*\*