National University of Singapore

TIC1001—Introduction to Computing and Programming I

Semester 1, 2017/2018

Time allowed: 2 hours

---

1. Please write your Student Number only. Do not write your name.
2. The assessment paper contains **SEVEN (7) questions** and comprises **FOURTEEN (14) pages** including this cover page.
3. Weightage of questions is given in square brackets. The maximum attainable score is 100.
4. This is a **CLOSED** book assessment, but you are allowed to bring **ONE** double-sided A4 sheet of notes for this assessment.
5. Five additional minutes of reading time will be given before the start of the assessment. You may read the paper but are not allowed to write anything during this time.
6. Write all your answers in the space provided in this booklet.
7. You are allowed to write with pencils, as long as it is legible.
8. **Please write your student number below.**

STUDENT NO: | A | | | | | | | | |

(This portion is for the examiner's use only)

Question	Marks	Remarks
Q1	/ 30	
Q2	/ 18	
Q3	/ 12	
Q4	/ 9	
Q5	/ 16	
Q6	/ 12	
Q7	/ 3	
Total	/ 100	

This page is intentionally left blank.

It may be used as scratch paper.

Question 1: C Expressions [30 marks]

There are several parts to this question which are to be answered independently and separately. Each part consists of a fragment of C code. Write the **exact output** produced by the code in **the answer box**. If an error occurs, or it enters an infinite loop, state and explain why.

You may show workings **outside the answer box** in the space beside the code. Partial marks may be awarded for workings if the final answer is wrong.

Assume that all appropriate preprocessor directives e.g., `#include <stdio.h>`, etc. have already been defined.

A. [5 marks]

```
int i ;
for (i = 1; i % 5; i += 3) {
    printf("%d ", i);
}
printf("\nDone %d",i);
```

B. [5 marks]

```
char s[] = "pineapple";
for (int i = 0; s[i]; i++) {
    int j = strlen(s)-1-i;
    char temp = s[i];
    s[i] = s[j];
    s[j] = temp;
}
printf("%s", s);
```

C. [5 marks]

```
int * foo(int *a, int n) {
    a[n] = a[(n*2)%7];
    return a;
}

int arr[] = {1, 2, 3, 4, 5, 6, 7};
foo(foo(foo(arr, 4), 2), 6);
for (int i = 0; i<7; i++)
    printf("%d ", arr[i]);
```

D. [5 marks]

```
int i = 1, j = 10;
do {
    if (i % j)
        i++;
    if (j % i)
        break;
    else
        j--;
} while (i < j);
printf("%d %d", i, j);
```

E. [5 marks]

```
int bar(char x, char *y) {
    int i = 0;
    while (y[i] != x)
        i++;
    return i;
}
char *s = "TIC1001";
printf("%d ", bar('1', s));
printf("%d ", bar(0, s));
```

F. [5 marks]

```
int g(int *a) {
    *a *= 2;
    return *a;
}
int f(int *a) {
    return g(a) + 3;
}
int a = 5;
int b = f(&a);
printf("%d %d", a, b);
```

Question 2: Stock Market [18 marks]

The consecutive daily closing prices of a stock can be represented with an array of integers. For example, an array `{5, 10, 4, 6, 10, 12, 8, 8, 3, 5, 6}` means that the price of the stock is 5 on the first day, 10 on the second day and so on.

A. Suppose we are interested in tracking the difference in the price of the stock from the preceding day. The function `void price_diff(int diff[], int prices[], int size)` takes as inputs two arrays, diff and prices, of the given size. The function should populate the array `diff` with the daily difference in stock prices.

For example, if `prices = {5, 10, 4, 6, 10, 12, 8, 8, 3, 5, 6}`, then the contents of `diff` will be `{0, 5, -6, 2, 4, 2, -4, 0, -5, 2, 1}`. You may assume that `diff` has been zeroed. [6 marks]

```
void price_diff( int diff[], int prices[], int size ) {




}
```

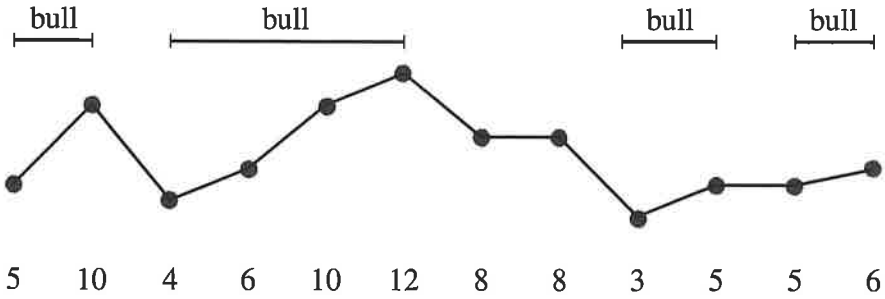
B. Some analysists want to know the number of days where the stock price is higher than the day before. Implement the function `int num_rises(int prices[], int size)` which takes in an array of prices of the given size, and returns the number of days where the price has risen from the day before. [6 marks]

```
int num_rises(int prices[], int size) {




}
```

C. Suppose a bull is the largest span where the price of the stock is increasing every consecutive day, i.e. a bull will begin when the stock price start to rise and ends when it stops rising. The figure below shows the number of bulls for the given daily stock prices.



Implement a function `int num_bulls(int prices[], int size)` that takes in an array of daily stock prices of length size, and returns the number of bulls found in the prices. [6 marks]

```
int null_bulls( int prices[], int size ) {



}
```

Question 3: Digit Sums [12 marks]

A. The digit sum of an integer is the sum of all its digits. The sign of the integer is ignored. In other words, the digit sum of negatives integers is the same as the sum of its absolute value.

Implement the function `int digit_sum(int n)` that returns the digit sum of the integer n. [6 marks]

```
int digit_sum(int n) {




}
```

B. A digital root, or also known as a repeated digit sum, of an integer is the single digit value obtained by repeatedly summing the digits, using the result from the previous iteration to compute the digit sum. This process continues until a single-digit number is reached.

For example, the digital root of 65536 is 7. Because $6+5+5+3+6=25$ and then $2+5=7$. As before, the digital root of negative integers is the same as that of its absolute value.

Implement the function `int digital_root(int n)` that returns the digital root of the input n. You may reuse the function `digit_sum` defined above. [6 marks]

```
int digital_root( int n ) {




}
```

Question 4: Operating System [9 marks]

Indicate whether the following statement is **True** or **False**, then give a **simple explanation/elaboration**.

A. On a single core processor, there are multiple processes running simultaneously for a multi-tasking system. [3 marks]

True / False.

B. On a single core processor, there are multiple processes in the **physical memory**. [3 marks]

True / False.

C. A file is always stored as a contiguous region on the secondary storage. [3 marks]

True / False.

Question 5: Cache & Virtual Memory [16 marks]

This question uses the following array:

```
#define SIZE 12
double dArray[SIZE];
```

Note that a double value takes **8 bytes** in the memory. For simplicity, you can assume element `dArray[0]` is at address 0.

Consider the following two different caches below:

Fully Associative One (FA1)	Fully Associative Two (FA2)
Total Size = 64 bytes	Total Size = 64 bytes
Cache Block Size = 32 bytes	Cache Block Size = 16 bytes

A. Give the block number for element `dArray[7]` for cache FA1 and FA2: [2 marks]

FA1: block _____

FA2: block _____

.....

For each of the code fragments below, give the total number of cache hit for cache FA1 and FA2:

B. [4 marks]

```
for (int i = 0; i < SIZE; i++){
    dArray[i] = 3.14159 * i;
}
```

FA1: Number of cache hit = _____

FA2: Number of cache hit = _____

C. [4 marks]

```
for (int i = 0; i < SIZE; i += 2){
    dArray[i] = 3.14159 * i;
}
for (int i = 1; i < SIZE; i += 2){
    dArray[i] = 3.14159 * i;
}
```

FA1: Number of cache hit = _____

FA2: Number of cache hit = _____

.....

Let us now consider paging for the following questions. You can assume a page size of **32 bytes** and the following page table:

Page Number	Frame Number
0	???
1	4
2	0
...	...

D. What is the Physical Address for element `dArray[7]` and `dArray[8]`? [4 marks]

`dArray[7]` is at _____

`dArray[8]` is at _____

E. What is the frame number for page 0 if element `dArray[2]` has the physical address 208? [2 marks]

Frame number is _____.

Question 6: Database [12 marks]

The table `Product` contains items sold in a supermarket. Its contents is as follow:

Name	Category	Manufacturer	Barcode	Price
Putter	Golf	Winfirst	4015	5.50
BHC Marker	Stationery	Kat	5903	26.85
6CM Canister	Tool	Chux	2639	91.40
Heavy Visor	Tool	Burgen	3854	65.30
Carpenter Pencil	Stationery	Kat	2341	22.75
Metal Mee	Food	Cadbury	5881	10.50
Rice Pasta	Food	Dynamo	3210	0.50
Writing Set	Stationary	Hahn	1506	47.95
Iron 5	Golf	Wilsoff	6139	48.35
Star Cover	Stationery	Hahn	1249	13.25
Highlander	Food	Chux	7501	72.85
Binoculars	Daily	Dynamo	7389	98.35
Card Holder	Stationery	Dynamo	6118	9.95
Mini Tool	Tool	Starburst	8213	24.25
Insence Shot	Tool	Dynamo	1841	33.60

A. What is the result for this SQL query? [3 marks]

```
SELECT Category FROM Product WHERE Price > 50;
```

B. What is the result for this SQL query? [3 marks]

```
SELECT Name, Barcode FROM Product WHERE Category="Stationery" AND Price < 20;
```

C. Give the SQL statement to create the table `Product` above. [3 marks]

D. Give the SQL statement to insert the first product "Putter" into the table. [3 marks]

Question 7: 42 and the Meaning of Life [3 marks]

Either: (a) explain how you think some of what you have learnt in TIC1001 will be helpful for you for the rest of your life and/or studies at NUS; or (b) tell us an interesting story about your experience with TIC1001 this semester. [3 marks]

—END OF QUESTIONS—

Scratch Paper

— END OF PAPER —