

Exercise 1

(5 Marks)

Sequential Algorithm

A game of luck is meant to find a person's lucky number. The lucky number is simply calculated by dividing the age of the person by 4, then adding 2 to it. Write a an algorithm that takes as input the year of birth of the person and calculates and prints the person's age, as well as her lucky number. **Hint:** Assume that we are in year 2006.

Solution:

```
get year
set age to 2006 - year
set lucky to age/4 + 2
print age
print lucky
```

Exercise 2

(7 Marks)

Conditional Algorithm

We would like to have an algorithm that prints out whether a current student is in elementary (1st - 5th), middle (6th - 8th), or high school (9th - 12th). Does the following algorithm correctly do this? Briefly explain why or why not. If not, correct the algorithm.

```
get grade
if (grade <= 5) then
    print "this student is in elementary school"
endif
if (grade <= 8) then
    print "this student is in middle school"
endif
if (grade <= 12) then
    print "this student is in high school"
endif
```

Solution:

The algorithm is not correct. For example for grade = 3 the algorithm will print out

```
this student is in elementary school
this student is in middle school
this student is in high school
```

Correct algorithm:

```
get grade
If (grade <= 5) then
    print "this student is in elementary school"
else if (grade <= 8) then
    print "this student is in middle school"
    else if (grade <= 12) then
        print "this student is in high school"
    endif
endif
endif
```

Exercise 3

(6 Marks)

Conditional Algorithm

Write an algorithm that given two integer numbers m and n checks if one of them is a multiple of the other or not. Use only conditional operations. **Hint:** Use the modulus (%) operator to check whether a number is divisible by another number.

For example, if $n=8$ and $m=64$ the program should output `true` and if $n=7$ and $m=6$ the program should output `false`.

Solution:

```
get m
get n
if ((n%m == 0) or (m%n == 0))
    print "true"
else
    print "false"
endif
```

Exercise 4

(10 Marks)

Mark the correct answer in the following (**only one choice**):

- 1) If a program compiles fine and runs fine but it produces incorrect answers, then the program suffers from

- ☐ Syntax error
☐ Run time error
☒ Logic error
☐ None of the above

- 2) What does the following code fragment write to the monitor?

```
int sum = 14;  
if ( sum < 20 )  
    { System.out.print("Under "); }  
else  
    { System.out.print("Over "); }  
System.out.println("the limit.");
```

- ☐ Under
☐ Over
☒ Under the limit.
☐ Over the limit.
☐ None of the above

- 3) The following algorithm calculates the **correct** letter grade 'A', 'B', 'C', 'D', or 'F'

```
if (score >= 90)  
    letter = 'A';  
if (score >= 80)  
    letter = 'B';  
if (score >= 70)  
    letter = 'C';  
if (score >= 60)  
    letter = 'D';  
else  
    letter = 'F';
```

- ☐ for all cases
☐ only for score >= 60
☐ only for score > 90
☒ only for score < 70
☐ This code will not work correctly under any circumstances

- 4) Examine the following code fragment:

```
int i = 1;  
while (i <= 6) {  
    System.out.print (2*i + " ");  
    i = i + 1;  
}
```

What is output to the monitor?

- ☐ 1 2 3 4 5 6
☐ 2 4 6 8 10
☒ 2 4 6 8 10 12
☐ 1 2 3 4 5
☐ None of the above

5) Examine the following code fragment:

```
int n = 3;
while (n != 1) {
    System.out.print(n + " ");
    if (n%2 == 0) {
        n = n / 2;
    } else {
        n = n*3 + 1;
    }
}
```

What is output to the monitor?

- | | |
|-------------------------------------|-------------------|
| <input type="checkbox"/> | 3 10 5 16 8 4 2 1 |
| <input type="checkbox"/> | 16 8 4 2 3 10 5 |
| <input type="checkbox"/> | 1 2 5 8 |
| <input checked="" type="checkbox"/> | 3 10 5 16 8 4 2 |
| <input type="checkbox"/> | 8 4 2 1 10 3 |
| <input type="checkbox"/> | None of the above |

6) The order of magnitude of the following algorithm is

```
set i to 1
while (i <= 100)
{
    print i
}
```

- | | |
|-------------------------------------|-------------------|
| <input type="checkbox"/> | $O(n)$ |
| <input type="checkbox"/> | $O(100)$ |
| <input checked="" type="checkbox"/> | $O(1)$ |
| <input type="checkbox"/> | $O(i)$ |
| <input type="checkbox"/> | None of the above |

Exercise 5

(8 Marks)

Iterative Algorithm

Write an algorithm that, given an integer number, calculates and prints the sum of its digits. For Example, if the number is 173, the sum printed should be 11 ($1+7+3$). Your algorithm should work for any integer value.

Solution:

```
get num
set sum to 0
while (num>0)
{
    set r to num%10
    set num to INT(num/10)
    set sum to sum + r
}
print sum
```

Exercise 6

(14 Marks)

Iterative Algorithm – Tracing

Given the following algorithm, where A_1, A_2, \dots, A_n is a list of integers

```

get A1, A2, ..., An
get n
set i to 0
while ( i < INT(n/2) )
{
    set temp to An-i
    set An-i to Ai+1
    set Ai+1 to temp
    set i to i + 1
}
print A1, A2, ..., An

```

- a) What is the output of the algorithm for the following list. Use a tracing table.

10 12 14 5 6 8

Solution:

| temp | i | Ai+1 | An-i |
|------|---|------|------|
| | 0 | 10 | 8 |
| 8 | 1 | 8 | 10 |
| 6 | 2 | 6 | 12 |
| 5 | 3 | 5 | 14 |

8 6 5 14 12 10

- b) What is the output of the algorithm for any list of size n ?

Solution:

The algorithm reverses the order of the list.

- c) Find the total number of executed operations. Show your workout.

Solution:

```

get A1, A2, ..., An
get n
set i to 0 ----- 1 operation    --> executed once
while ( i < INT(n/2) ) ----- 1 operation    --> INT(n/2) + 1 repetitions
{
    set temp to An-i ----- 1 operation    --> INT(n/2) repetitions
    set An-i to Ai+1 ----- 1 operation    --> INT(n/2) repetitions
    set Ai+1 to temp ----- 1 operation    --> INT(n/2) repetitions
    set i to i + 1 ----- 1 operation    --> INT(n/2) repetitions
}
print A1, A2, ..., An ----- 1 operation    --> executed once

```

Total number of executed operation $5 * \text{INT}(n/2) + 3$

- d) Determine the order of magnitude of the algorithm.

Solution:

$O(n)$

Exercise 7

(8 Marks)

Iterative Algorithm

Write an algorithm that given an integer number determines whether the number is a prime number.

A *prime number* is a number that can only be divided by itself and 1. For example, 11 is a prime number because it can be only divided by 1 and 11. However, 8 is not a prime number since it can be divided by 1, 2, 4 and 8. **Hint:** Use the modulus (%) operator to check whether a number is divisible by another number.

Solution:

```
get number
set result to true
if (number = 1) then
    set result to true
else
    if (number = 0)
        set result to false
    endif
endif
set i to 2
while (i < number)
{
    if ((number%i) = 0) then
        result = false
    endif
    set i to i + 1
}
print result
```

Exercise 8 Bonus Exercise

(8 Marks)

Iterative Algorithm – Lists

Write an algorithm that given an **ordered** list of integers A_1, A_2, \dots, A_n prints the elements in the list that are repeated. If some elements occur more than twice, then these elements should be printed only once.

For example, for the list

1 1 1 1 4 6 7 7 8

your algorithm should print

1 7

Solution:

```
get n
get A1, ..., An
set flag to A1 - 1
set i to 1
while (i < n )
{ if (flag <> Ai)
  then
    if (Ai = Ai+1) then
      set flag = Ai
      print Ai
      set i to i + 2
    else
      set i to i + 1
    endif
  else
    set i to i+1
  endif
}
```

Extra Sheet

Extra Sheet

Extra Sheet