|  |  |  |
| --- | --- | --- |
| Computing Fundamentals | 2020-2021 | |
| This document contains assessment criteria that contributes 28 credits out of 100 of your work for Computing Fundamentals. The summative assessment is due each week | | Level C |

# Contents

[Contents i](#_bookmark0)

[Introduction 1](#_bookmark1)

[Schedule 1](#_bookmark2)

[Academic Misconduct 1](#_bookmark3)

[If your tutor(s) suspect academic misconduct 2](#_bookmark4)

[Assessment Regulations 3](#_bookmark5)

[Week 0 (part of your assessment) 4](#_bookmark6)

[Synopsis 4](#_bookmark7)

[Exercise 4](#_bookmark8)

[Alternative Exercise for Students with Macs 4](#_bookmark9)

[Submission guidelines 4](#_bookmark10)

[Marking Guidance 4](#_bookmark11)

[Week 1 (for reinforcement of knowledge and not assessed) 5](#_bookmark12)

[Synopsis 5](#_bookmark13)

[Exercise 5](#_bookmark14)

[Submission guidelines 5](#_bookmark15)

[Marking Guidance 5](#_bookmark16)

[Week 2 (part of your assessment) 6](#_bookmark17)

[Synopsis 6](#_bookmark18)

[Exercise 6](#_bookmark19)

[Submission guidelines 6](#_bookmark20)

[Marking Guidance 6](#_bookmark21)

[Week 3 (for reinforcement of knowledge and not assessed) 7](#_bookmark22)

[Synopsis 7](#_bookmark23)

[Exercise 7](#_bookmark24)

[Submission guidelines 7](#_bookmark25)

[Marking Guidance 7](#_bookmark26)

[Week 4 (part of your assessment) 8](#_bookmark27)

[Synopsis 8](#_bookmark28)

[Exercise 8](#_bookmark29)

[Submission guidelines 8](#_bookmark30)

[Marking Guidance 8](#_bookmark31)

[Week 5 (for reinforcement of knowledge and not assessed) 9](#_bookmark32)

[Synopsis 9](#_bookmark33)

[Exercise 9](#_bookmark34)

[Submission guidelines 9](#_bookmark35)

[Marking Guidance 9](#_bookmark36)

[Week 6 (part of your assessment) 10](#_bookmark37)

[Synopsis 10](#_bookmark38)

[Exercise 10](#_bookmark39)

[Submission Guidelines 10](#_bookmark40)

[Marking Guidance 10](#_bookmark41)

[Week 7 (for reinforcement of knowledge and not assessed) 11](#_bookmark42)

[Synopsis 11](#_bookmark43)

[Exercise 11](#_bookmark44)

[Submission guidelines 11](#_bookmark45)

[Marking Guidance 11](#_bookmark46)

[Week 8 (part of your assessment) 12](#_bookmark47)

[Synopsis 12](#_bookmark48)

[Exercise 12](#_bookmark49)

[Submission guidelines 12](#_bookmark50)

[Marking Guidance 12](#_bookmark51)

[Week 9 (for reinforcement of knowledge and not assessed) 13](#_bookmark52)

[Synopsis 13](#_bookmark53)

[Exercise 13](#_bookmark54)

[Submission guidelines 13](#_bookmark55)

[Marking Guidance 13](#_bookmark56)

[Week 10 (part of your assessment) 14](#_bookmark57)

[Synopsis 14](#_bookmark58)

[Exercise 14](#_bookmark59)

[Submission Guidelines 14](#_bookmark60)

[Marking Guidance 14](#_bookmark61)

[Week 11 (for reinforcement of knowledge and not assessed) 15](#_bookmark62)

[Synopsis 15](#_bookmark63)

[Exercise 15](#_bookmark64)

[Marking Guidance 15](#_bookmark65)

[Week 12 (part of your assessment) 16](#_bookmark66)

[Synopsis 16](#_bookmark67)

[Exercise 16](#_bookmark68)

[Submission Guidelines 16](#_bookmark69)

[Marking Guidance 16](#_bookmark70)

[Week 13 (for reinforcement of knowledge and not assessed) 17](#_bookmark71)

[Synopsis 17](#_bookmark72)

[Exercise 17](#_bookmark73)

[Submission Guidelines 17](#_bookmark74)

[Marking Guidance 17](#_bookmark75)

[Week 14 (part of your assessment) 18](#_bookmark76)

[Synopsis 18](#_bookmark77)

[Exercise 18](#_bookmark78)

[Submission Guidelines 18](#_bookmark79)

[Marking Guidance 18](#_bookmark80)

[Week 15 (for reinforcement of knowledge and not assessed) 19](#_bookmark81)

[Synopsis 19](#_bookmark82)

[Exercise 19](#_bookmark83)

[Submission Guidelines 20](#_bookmark84)

[Marking Guidance 20](#_bookmark85)

[Week 16 (part of your assessment) 21](#_bookmark86)

[Synopsis 21](#_bookmark87)

[Exercise 21](#_bookmark88)

[Submission Guidelines 21](#_bookmark89)

[Marking Guidance 21](#_bookmark90)

[Week 17 (for reinforcement of knowledge and not assessed) 22](#_bookmark91)

[Synopsis 22](#_bookmark92)

[Exercise 22](#_bookmark93)

[Submission Guidelines 22](#_bookmark94)

[Marking Guidance 22](#_bookmark95)

[Week 18 (part of your assessment) 23](#_bookmark96)

[Synopsis 23](#_bookmark97)

[Exercise 23](#_bookmark98)

[Submission Guidelines 23](#_bookmark99)

[Marking Guidance 23](#_bookmark100)

# Introduction

Each week you will be assessed on the subject material for Structured Problem Solving. This will be in the form of small portfolios and will require submitting the week after it is set, for example week 2 assessment will be submitted in week 3. You will not be able to submit assessment once the opportunity has passed unless you have mitigating evidence, i.e. week 2 assessment cannot be submitted in week 4.

You will be graded for each assessment as: not submitted or very little detail, attempted, good attempt or completely correct (0-4 Marks) unless stated otherwise. Moodle will show a mark out of 100 for each week. For example, a mark of 4 is 100, 3 is 75, 2 is 50, 1 is 25. This assessment constitutes 1/3rd of your overall mark for the Computing Fundamentals course.

Although you will submit each assessed portfolio each week you will also submit all the portfolios again as a final completed document. This document is the amalgamation of all the portfolios that you completed.

The portfolios can be thought of as homework, they are to be completed outside of scheduled class time and you are to work individually. Your seminar tutor may allow you to work on portfolio assessments during the seminar class, but only once all seminar activities for that week have been fully completed.

Each portfolio is to be word processed and presented professionally. If the work is illegible you will be awarded a mark of 0.

## Schedule

|  |  |  |
| --- | --- | --- |
|  | Assessment Set | Assessment Due |
| Week 0 | 30/09/2020 | 14/10/2020 |
| Week 1 | OPTIONAL | OPTIONAL |
| Week 2 | 14/10/2020 | 28/10/2020 |
| Week 3 | OPTIONAL | OPTIONAL |
| Week 4 | 28/10/2020 | 25/11/2020 |
| Week 5 | OPTIONAL | OPTIONAL |
| Week 6 | 25/11/2020 | 09/12/2020 |
| Week 7 | OPTIONAL | OPTIONAL |
| Week 8 | 09/12/2020 | 27/01/2021 |

|  |  |  |
| --- | --- | --- |
| Week 9 | OPTIONAL | OPTIONAL |
| Week 10 | 27/01/2021 | 10/02/2021 |
| Week 11 | OPTIONAL | OPTIONAL |
| Week 12 | 10/02/2021 | 01/03/2021 |
| Week 13 | OPTIONAL | OPTIONAL |
| Week 14 | 03/03/2021 | 17/03/2021 |
| Week 15 | OPTIONAL | OPTIONAL |
| Week 16 | 17/03/2021 | 28/04/2021 |
| Week 17 | OPTIONAL | OPTIONAL |
| Week 18 | 17/03/2021 | 28/04/2021 |

## Academic Misconduct

The University takes academic misconduct very seriously and could lead to the studies of a student being terminated. This section of the document identifies two types of academic misconduct and offers guidance on how to avoid such cases.

Plagiarism is presenting somebody else’s work as your own. Most common types of plagiarism are

copying and pasting text from the Internet or other sources, other types of plagiarism are presenting

others ideas as your own. To avoid plagiarism, you should not copy text from other sources verbatim. In some cases, you may want to quote a particular piece of text, you can do this by including the text in quotation marks, citing the quote and then providing the full reference for the citation in a reference list. To avoid being accused of passing others ideas as your own, any text you paraphrase should also be cited and referenced.

Collusion is working together with one or more persons on the same piece of work and then submitting the work as your own. To avoid collusion, you should complete the assessment on your own. Never share a copy of your work with another student, even if it is a draft or the submission deadline has passed.

## If your tutor(s) suspect academic misconduct

Should your tutor(s) suspect academic misconduct they will gather evidence that supports the suspicion, such evidence may include TurnItIn reports and the original articles. When suspected of collusion the work of all parties is retained as evidence.

Any such evidence is then passed to the Faculty Senior Academic Advisor (FSAA) who will consider the evidence presented. If the FSAA also suspects that academic misconduct has occurred, you will receive a letter from the FSAA inviting you to attend an interview to discuss the allegation.

The meeting will consist of a member of staff who marked your submission, the FSAA, yourself and another person of your choice (optional) to support you. At this meeting the FSAA will explain the remit of the meeting and the regulations. The tutor will explain to you where the suspected academic misconduct has occurred and ask you for explanations.

The FSAA will form a judgment based on the evidence and meeting, this judgment will be presented to the Board of Examiners. The Board of Examiners will make the final decision.

An initial case of academic misconduct normally involves the Head of Department issuing a warning to the student. At level C the original work receives a mark of 0PL and the student is given the opportunity to represent the work within a reasonable timescale, having addressed the issues and made the necessary amendments. The represented work shall be marked on a bare pass or fail basis.

A second case of academic misconduct normally involves terminating the studies of the student.

Cases of academic misconduct are recorded on your student record and do not expire as you progress from level C to I or I to H. Therefore, a first case at level I will be treated as the second case if you already have a record of academic misconduct at level C.

Please do consult the Harvard referencing guide which is located on Moodle for further information on how to avoid being suspected of academic misconduct.

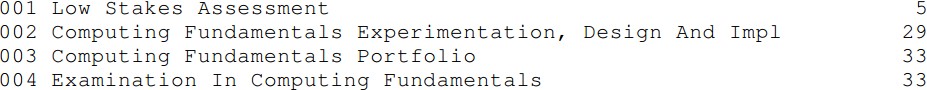
This is an overview of the regulations. The full regulations can be located here:

https://[www.hope.ac.uk/media/gateway/studentgateway/supportandwellbeing/studentadministrat](http://www.hope.ac.uk/media/gateway/studentgateway/supportandwellbeing/studentadministrat) iondocuments/t4\_111329\_Media.pdf

# Assessment Regulations

Ideally you will want to submit each and every piece of work with the aim of achieving the highest possible grade. The University pass grade at undergraduate level is 40.

Your assessment lines for Computer Fundamentals look like:



and this document is for 003 Computing Fundamentals Portfolio (33% of the course).

In order to pass the course you need to pass every assessment line 1 (except 001 Low Stakes Assessment, also known as the Early Assessment). Therefore, you should submit all work ***before*** the deadline(s).

If you are unable to meet a deadline because of mitigating circumstances (and you can provide evidence for) you should speak with Dr. Thanapong, Dr. Ranasinghe or Dr. Secco who may be able authorize an extension up to one week (depending on evidence).

If you are unable to meet the deadline for any other reason you should speak with your personal tutor to discuss your situation.

### Only exercises falling on an even-numbered week (0 to 18 inclusive) are formally assessed and count toward your portfolio grade. Those falling on odd-numbered weeks (1 to 17 inclusive) are for practice, and will not automatically be marked by your tutor. However, you should make every effort to complete these, as they will contribute toward your education. If you want feedback on these non-assessed exercises, feel free to request this from your tutor.

Full details of assessment regulations can be found here:

[https://www.hope.ac.uk/gateway/supportandwellbeing/studentadministration/understandingyourd](https://www.hope.ac.uk/gateway/supportandwellbeing/studentadministration/understandingyourdegree/assessmentofstudentsregulations/) [egree/assessmentofstudentsregulations/](https://www.hope.ac.uk/gateway/supportandwellbeing/studentadministration/understandingyourdegree/assessmentofstudentsregulations/)

1 The University does allow for the compensation of a fail if the work achieved a mark between 35 and 39, and the overall aggregate mark is 40 or higher for that course.

# Week 0 (part of your assessment)

## Synopsis

The first assessment ensures that you understand how to use the JavaTrainer software that is provided.

## Exercise

Create a program that will trace out a staircase which goes from the **bottom** of the grid to the top.

## Alternative Exercise for Students with Macs

On a piece of paper, or in a spreadsheet program (such as Excel), draw a 3x3 grid of cells. Represent the “PieEater” with the an arrow, or the characters “-“, “|” or “/”, depending on which way it is facing. Empty cells should be left blank. The PieEater should start in the bottom left-cell. Write the sequence of commands that make the PieEater walk to the right once, turn left twice, walk up, then continue in this staircase-like pattern until it resides in the top-right cell facing toward the right.

Accompany each step with an updated version of your illustration. Remember to use the pendown() command, which would be used in Javatrainer to begin drawing a trail.

## Submission guidelines

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in JavaTrainer and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. If this is not clear at this stage please don’t worry. We will not penalize for this point at this stage of the course.

An example of points 1 and 2 is in next week’s assessment on the next page. Submit the assessment to Moodle using the link Staircase Submission

## Marking Guidance

The program is entirely correct. [4 Marks] or

The program is mostly correct (i.e. wrong start position). [3 Marks] or

The program has been attempted but some issues [2 Marks] or

Attempted but major issues [1 Mark]

# Week 1 (for reinforcement of knowledge and not assessed)

## Synopsis

This session builds on the skills you learnt in week 0 and assesses your understanding of the for

loop.

## Exercise

Type in the program below into JavaTrainer; amend this program to make PieEater walk around the grid ending up where he started leaving a trail as he walks.

createpieeater(); int i;

for (i=1 ; i<3 ; i++)

{

walk();

}

**Alternative Exercise for Mac Users**

Study the above code and research the use of functions in coding, as well as variables and “for”

loops. The start of the loop has three terms in brackets. List what each of these indicate.

## Submission guidelines

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in JavaTrainer and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces. If this is not clear at this stage please don’t worry. We will not penalize for this point at this stage of the course.

Submit the assessment Moodle using the link BoundaryWalk Submission.

## Marking Guidance

The program is correct and both a while loop and a for loop used [4 Marks] or The program is correct and good use of loops. [3 Marks] or

The program is correct but no additional loop used. [2 Marks] or

Attempted but some issues [1 Mark]

# Week 2 (part of your assessment)

## Synopsis

This week we test your understanding of problem solving and representing that problem as an algorithm in Structured English.

## Exercise

Given the following criteria:

**Variables** money := 100, marbleCost := 12, counter := 0

**Limits** money should not be a negative value

**Result** Display the value of counter once the algorithm has completed. In structured English you can use the keyword **display** and the variable name to display the value, for example: display counter

Complete an algorithmic design that will purchase a marble **while** money is greater or equal to marbleCost. The solution should keep purchasing a marble until the marbleCost is greater than money. Each time a marble is purchased you should deduct the marbleCost from money.

## Submission guidelines

There are many portfolio exercises where we ask you to submit structured English. We ask you to adhere to house formatting rules when submitting structured English which are the same as the house rules for formatting code. These are:

1. All structured English should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces.

## Marking Guidance

Program is correct, initialisation correct, loops correct, variable use correct [4 Marks] or The program is correct and good use of loops. [3 Marks] or

The program is correct but could have been better optimized. [2 Marks] or

Attempted but some issues [1 Mark]

**Week 3** (for reinforcement of knowledge and not assessed)

## Synopsis

This week we assess your understanding of trace tables. Trace tables allow us to trace through a given algorithmic design and keep track of variable values and conditional statement evaluations.

## Exercise

Create a trace table with the appropriate headings and trace through the following Structured English.

x := 5

y:= 10

while (x < y) begin

x:=x+2 y:=y+1

if (x not < y) then begin

y := 50

x := 50

end

end

step 1

step 2

step 3

step 4

step 5

step 6

step 7

step 8

## Submission guidelines

The document should be word processed and make use of a table. The submission link is on moodle and is labelled Trace Table Submission.

## Marking Guidance

The trace table is entirely correct [4 Marks] or

The trace table contains less than 5 errors [3 Marks] or

Not Looping back correctly at step 3 but fine until this point [2 Marks] or

Attempted but some issues [1 Mark]

# Week 4 (part of your assessment)

## Synopsis

This week we further assess your understanding of algorithmic design by looking at a given scenario and the requirements or operations of that scenario.

## Exercise

Write an algorithm to accomplish the following:

The temperature inside a pottery kiln is sampled periodically during the firing process. When the temperature goes outside a critical range, a light flashes on the operator's control panel. A red light indicates that a dangerously high temperature has been reached, whilst a blue light indicates the opposite.

This type of emergency prompts the operator to go and follow a set procedure involving three controls, labelled A, B and C.

If the temperature is critically low, control A must be set to " Open Valve", control B must be set to "Increase" and control C must be set to "Emergency Low"

If the temperature is critically high, A is set to "Close Valve", B is set to "Decrease" and C is set to "Emergency High".

When the lights stop flashing the controls are set to their normal positions.

## Submission guidelines

There are many portfolio exercises where we ask you to submit structured English. We ask you to adhere to house formatting rules when submitting structured English which are the same as the house rules for formatting code. These are:

1. All structured English should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces. The submission link is on Moodle and is labelled PotteryKiln Submission.

## Marking Guidance

The solution is correct [4 Marks] or

The solution is correct (some misunderstanding with := and =) [3 Marks] or

The solution is mostly correct, further refinement needed. [2 Marks] or

Attempted but some issues [1 Mark]

**Week 5** (for reinforcement of knowledge and not assessed)

## Synopsis

This week we strengthen our understanding of step wise refinement and look at using functions. Functions can accepts values, return a value or both. Functions can also be used as the condition for conditional statements.

## Exercise

A sensor is to be installed by the main London to Scotland railway line at Milton Keynes. The sensor will be installed in the signal box to the south of the station and will satisfy the following specification:

* + The sensor will only operate during daylight hours
  + the sensor will classify the length of each train according to one of the categories: Long, Medium or Short
  + The sensor will keep a running total of the number of trains in each category
  + The sensor will only respond to north bound trains

Design, using suitable pseudo-code/structured English, an algorithm for the above specification using the standard problem solving constructs and the following operations and variables:

**Direction(train)** returns the value NORTH or SOUTH **Length(train**) returns a value Long, Medium or Short **L, M and S** are variables which can store integer values **Day\_Light** can be True or False

## Submission guidelines

There are many portfolio exercises where we ask you to submit structured English. We ask you to adhere to house formatting rules when submitting structured English which are the same as the house rules for formatting code. These are:

1. All structured English should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces. The submission link is on Moodle and is labelled TrainSensor Submission.

## Marking Guidance

The solution is correct. [4 Marks] or

The solution is correct, some misunderstanding with instructions [3 Marks] or The solution is mostly correct, further refinement needed. [2 Marks] or

Attempted but some issues [1 Mark]

# Week 6 (part of your assessment)

## Synopsis

This portfolio assesses your understanding of the Java concepts you met in the first two weeks of the course.

## Exercise

Create the algorithmic design in structured English that will accept a sequence of numbers, terminated by 0 and display the sum of the numbers, the average of the numbers, the minimum number and the maximum number. Then code the design.

## Submission Guidelines

There are many portfolio exercises where we ask you to submit structured English and code. We expect you to first problem solve and create an algorithmic design. Then program and test the solution in the programming environment, once working, copy and paste the code into a word document after the structured English.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link Calculate Submission

## Marking Guidance

The program is correct [4 Marks] or

The program is mostly correct but does not display all required values [3 Marks] or Attempted but some issues [2 Marks] or

Only the code or the structured English submitted (not both) [1 Mark]

**Week 7** (for reinforcement of knowledge and not assessed)

## Synopsis

This week you are to create an algorithmic design from a less well defined problem.

## Exercise

1. Create the algorithmic design that will make jelly!

Pour the first packet of jelly powder into a large bowl and while whisking, add the hot water (250ml). Whisk until the powder has dissolved. Slowly add the cold water (250ml) and whisk once more. Repeat exactly the same process with the second packet of jelly.

Generously ladle the jelly syrups into separate glass bowls. Place the jelly in the fridge and chill for 2-4 hours.

Remove the jelly from the fridge when fully set and serve with cream or ice cream.

## Submission guidelines

There are many portfolio exercises where we ask you to submit structured English. We ask you to adhere to house formatting rules when submitting structured English which are the same as the house rules for formatting code. These are:

1. All structured English should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces. The submission link is on Moodle and is labelled Jelly Submission.

## Marking Guidance

The solution is correct. [4 Marks] or

The solution is mostly correct, further refinement needed. [3 Marks] or

Attempted but some issues [2 Marks] or

Attempted but many issues [1 Mark]

# Week 8 (part of your assessment)

## Synopsis

More problem solving and structured English this week with the additional requirement of creating a

**flowchart** that represents your solution graphically. This assessment is based on a digital clock.

## Exercise

Given the following criteria:

**Variables** S := 0, M:= 0, H:=0, ON:=True

**Limits** S can be 0-60, M can be 0-60, H can be 0-24

**Conditions** S will count from 0 to 60 continually

When S becomes 60 it will be set to 0 and 1 will be added to M. When M becomes 60 it will be set to 0 and 1 will be added to H. When H becomes 24 it will be set to 0.

Create an algorithmic design for the above criteria and represent the solution in Structured English and flowchart form. The program should terminate when ON becomes equal to false.

## Submission guidelines

All diagrams should be clear. All conditional statements should be labelled true and false. You should use the drawing tools available in the application you are using (i.e. Microsoft Word) for a professional look.

There are many portfolio exercises where we ask you to submit structured English. We ask you to adhere to house formatting rules when submitting structured English which are the same as the house rules for formatting code. These are:

1. All structured English should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statements should be indented by 3 spaces. The submission link is on Moodle and is labelled Clock Submission.

## Marking Guidance

The solution is correct. [4 Marks] or

The solution is mostly correct, further refinement needed. [3 Marks] or

Attempted but some issues [2 Marks] or

Only the flowchart or the structured English submitted (not both) [1 Mark]

**Week 9** (for reinforcement of knowledge and not assessed)

## Synopsis

This week your assessment involves creating questions with solutions. This is designed to help with examination revision.

## Exercise

You should by now have a good understanding of Structured English, Flow Charts, Trace tables, Expressing a scenarios as a selection, Expressing the scenario as an iteration and expressing a scenario using all the three problem solving constructs.

This week instead of completing questions you will be writing the questions in the format you have seen in your seminar sessions for the last 4 weeks.

## Submission guidelines

Create the questions for the following structure:

1. Express the following algorithm design in flowchart form.
2. Create a trace table for the above structured English.
3. Express the following as a selection in Structured English.
4. Express the following as an iteration in Structured English.
5. Express the following instructions as Structured English

At the end of the document give solutions for the questions that you have created. Look at week 9 seminar exercises for examples for each of the above.

The submission link is on Moodle and is labelled Questions Submission.

## Marking Guidance

Structured English, Flow Chart and Trace Table questions complete. [4 Marks] or As above but no solutions. [3 Marks] or

Partially complete (i.e. two of the three questions) [2 Marks] or

As above but no solutions. [1 Mark]

# Week 10 (part of your assessment)

## Synopsis

These exercises are designed to reinforce your learning so far and to continue your revision for your examination.

## Exercise

Create a program in the processing environment that will mimic two dice. Call these dice1 and dice2. Both dice are traditional 6 faced cubed dice with numbers 1-6. The computer will roll the dice until snake eyes is achieved. Each roll the computer will generate statistics stating how close it was to achieving the goal. Example output from the program:

|  |  |  |  |
| --- | --- | --- | --- |
| Roll | Dice1 | Dice2 | Distance |
| 1 | 3 | 4 | 2+3=5 |
| 2 | 2 | 5 | 1+4=5 |
| 3 | 5 | 5 | 4+4=8 |
| 4 | 1 | 1 | SNAKE EYES! |

## Submission Guidelines

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in the programming environment and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link SnakeEyes Submission

## Marking Guidance

Correct formatting for house rules applied. [1 Mark]

Correct use of the random method. [1 Mark]

Program works correctly [1 Mark]

Format of the output from program matches example above (of course with different

values because they are randomly generated). [1 Mark]

**Week 11** (for reinforcement of knowledge and not assessed)

## Synopsis

We shift emphasis to data structures; a data structure is a particular way of storing and organizing data in a computer so that it can be used efficiently.

This week we test your awareness of the array data structure.

## Exercise

Produce a short report that: details an array, details a multidimensional array and explains the differences between the two (max 300 words). Your work should include references in Harvard Format. References are not part of the word count.

## Marking Guidance

Both 1D and Multidimensional arrays explained. [1 Mark]

The differences between a 1D and Multidimensional array present. [1 Mark]

Good use of citations and references [1 Mark]

References in Harvard Format [1 Mark]

Submit this to Moodle using the link Arrays Submission

# Week 12 (part of your assessment)

## Synopsis

This is the final week before Christmas and before we break we ask you for your feedback.

## Exercise

This assessment is to be completed over the Christmas break and presented the week you return.

You are to write a 500-1000 word reflective report on your studies for Computing Fundamentals so far. Your reflective report should be based on the areas that the national student survey questions, except instead of simply answering these questions you will give examples, this is particularly important for areas where you feel things could be improved upon. An example might be:

The teaching on my course is very good and the staff are excellent at explaining technical topics, for example when we were introduced to a for loop the tutor took time to explain the components of the loop until we all understood.

Or

The teaching on my course is ok but this could be improved upon, especially regarding the technical topics. When the tutor explained the for loop he went through this too quickly and left me confused. I’m still not entirely sure how to construct a loop without first looking at my notes.

We encourage you to be critical and not concern yourself with how the tutor may perceive your reflective report. The tutor is only interested in ensuring that they can improve in areas that may need improvement and to continue good practice in areas that you identify as good.

## Submission Guidelines

You are not to submit this work to Moodle as you have done previously. Instead you will submit to the departmental administrator who will record the submission. You do not have to put your name on the assessment if you would prefer not to.

## Marking Guidance

Submitted. [4 Marks]

**Week 13** (for reinforcement of knowledge and not assessed)

## Synopsis

Choosing the right data structure for your algorithm is essential for the optimization and functionality of that algorithm. The next three weeks will introduce you to data structures including: records, arrays, tables, trees, queues and stacks. Today we look at records and arrays.

This week we continue with data structures and in particular look at the java code in assigning values to the data structure, displaying a particular element within a data structure and stepping through the data structure an element at a time.

## Exercise

You are to write a program that will create a data structure. The data structure will store a list of

people’s forenames, e.g. people you know on your course. The contents of the data structure will be displayed on screen.

## Submission Guidelines

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in the programming environment and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link ArrayOfString Submission

## Marking Guidance

A suitable data structure has been created in the processing programming

environment and is correct. [1 Mark]

The data structure has been populated with data. [1 Mark]

A loop has been used that will loop through the data structure and print the value

of each element to screen. [1 Mark]

The loop correctly handles every array index [1 Mark]

# Week 14 (part of your assessment)

## Synopsis

Choosing the right data structure for your algorithm is essential for the optimization and functionality of that algorithm. The next three weeks will introduce you to data structures including: records, arrays, tables, trees, queues and stacks.

This week we continue with data structures and in particular look at applying an appropriate data structure to a given scenario, the data structures consist of Records, Arrays, Tables, Trees, Queues and Stacks.

## Exercise

You are to write a program similar to the one in the previous week, however this time the program will ask you to type the names that need to be stored. Once this is complete the program will display the names in alphabetical order and display the number of characters for each name and the number of vowels each name has.

## Submission Guidelines

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in the programming environment and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link ArrayOfStringAndChar Submission

## Marking Guidance

The program asks the user for input using a suitable input dialogue box. [1 Mark] The program successfully sorts the data within the data structure. [1 Mark]

Characters counted and the count displayed [1 Mark]

Vowels are counted and the count displayed [1 Mark]

**Week 15** (for reinforcement of knowledge and not assessed)

## Synopsis

Choosing the right data structure for your algorithm is essential for the optimization and functionality of that algorithm. .

We finish our introduction to data structures by looking at multi-dimensional arrays, these data structures are similar to records. Do you know the fundamental difference between multi- dimensional arrays and records?

This week we also begin object oriented concepts and programming.

## Exercise

A software developer has been commissioned to create a stock control system for a computer supplier in which personal computers (PCs) are to be modelled. He will need to be able to store for each type of PC the following information: the name of the PC, the web site address of the manufacturer, the number currently in stock and the selling price.

He will want to send messages to the PC object to make it carry out the following actions:

* set the total number of PCs of this type currently in stock to a given value;
* add a PC of this type to stock;
* remove a PC of this type from stock;
* reply with the number of PCs of this type currently in stock.

1. Draw an object diagram for a suitable PC object showing the attributes and protocol as described above with suitable names
2. Suppose a PC object called **PC1** is created with the name *Dell Inspiron 7500*. The manufacturer's web site address is [*www.dell.co.uk*.](http://www.dell.co.uk/) The supplier has initially a total of 25 of these PCs in stock. The selling price is £1100. Redraw your answer to part (a) to show the initial state of **PC1**.
3. Suppose the user of the stock control program carries out the following sequence of actions on **PC1**.

set the number of PCs of this type to 40 add a PC of this type to stock

add a PC of this type to stock remove a PC of this type from stock

Write a sequence of Java statements that would send suitable messages to **PC1** using the protocol you specified in part (a).

Redraw your answer to part (b) to show the resulting state of **PC1**.

## Submission Guidelines

You may find it useful to use the word processor table feature when creating the class diagrams. Alternatively you may find it easier to you a presentation or graphics program and insert the class diagram as an image. Whichever method you choose your work should be presented professionally and similar to the format used when we presented class diagrams to you in the seminar exercises.

Submit this to Moodle using the link ClassDiagram Submission

## Marking Guidance

Class Diagram presented. [1 Mark]

Initial state for PC1 presented. [1 Mark]

Resulting state for PC1 created. [1 Mark]

Object points to the Class Diagram [1 Mark]

# Week 16 (part of your assessment)

## Synopsis

This week you will be coding in Java, we will adopt Object Oriented approaches and manipulate existing code.

## Exercise

* 1. Create a copy of the *hopepicture* project by selecting **Project | Save As.**. and saving the project as *myPicture.* Using the existing code as a template to create a unique picture of your choice.

## Submission Guidelines

You are to take a screen shot of your picture and copy it into a word document. Do not capture the entire screen, we are only interested in the picture created by the program and the code to create the picture.

You are to submit your picture followed by your code in a word document.

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in the programming environment and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link MyPicture Submission

## Marking Guidance

A picture has been created. [1 Mark]

Picture has at least x2 objects. [1 Mark]

Picture has a least x5 objects. [1 Mark]

Correct use of Snipping Tool to present picture (i.e. not entire desktop) [1 Mark]

**Week 17** (for reinforcement of knowledge and not assessed)

## Synopsis

This week we aim to strengthen your understanding of object oriented concepts by producing a small report that details each of the concepts.

## Exercise

1. Write a short definition for each of the following terms: class, method, attribute, constructor and instance. You should also explain how the terms relate to each other and also include a diagram that shows the relationship between the terms.

## Submission Guidelines

You should include references for any work that is not your own.

Submit this to Moodle using the link ObjectOrientedConcepts Submission

## Marking Guidance

All terms explained. [1 Mark]

The relationship between the terms is explained. [1 Mark]

A diagram is presented that shows the relationships. [1 Mark]

Good use of citations and a reference list included (Harvard Format). [1 Mark]

# Week 18 (part of your assessment)

## Synopsis

This week you will be coding in Java, we will adopt Object Oriented approaches and manipulate existing code. The exercises are based on our friend PieEater and his world. We begin by creating PieEater, his world and simple messages that enable us to manipulate PieEaters behaviour such as: walk(), turnLeft() and turnRight(). You can be as creative as you wish, the world does not have to be an 8x6 grid, instead of walking why not hop or jump?

## Exercise

Create a new message called walkNTimes that will make PieEater walk the number of times specified by the user.

## Submission Guidelines

You should copy your walkNTimes method (not the entire program) into a word document.

There are many portfolio exercises where we ask you to submit code. We expect you to complete and test the program in the programming environment and then once working, copy and paste the code into a word document.

We ask you to adhere to house formatting rules when submitting code. These are:

1. All code should be formatted in font Courier New with a point size of 11.
2. Any paragraph spacing should be removed.
3. Any instructions inside a conditional statement should be indented by 3 spaces. Submit this to Moodle using the link walkNTimes Submission.

## Marking Guidance

The creation of a new message called walkNTimes. [1 Mark]

The message accepts an argument. [1 Mark]

The argument is used to control how many times a for loop iterates. [1 Mark] Completely correct [1 Mark]