

Student Name:

Computer Science – Preparatory Work

This course provides you with an opportunity to study for and gain an academic qualification in the field of Computer Science. In addition to learning about computer science topics and programming concepts, this course also utilises industry standard tools, such as Visual Studio and C#, to enable you to gain firm foundations for either further study or employment.

- 80% of the course is assessed through external examinations taken at the end of the second year.
- 20% is assessed through coursework which commences during the Summer term of the first year and is usually submitted during April of the second year.

The following tasks are to help get you organised, set-up and ready to start learning Computer Science. Please complete the tasks listed and hand in any work at the start of your first Computer Science lesson.

Task 1 – Folders and Dividers

Get organised by obtaining two lever arch files along with 10-part dividers to organise your work. You **don't** have to carry these to and from every lesson. You may find it convenient to carry something smaller to and from college and then you can transfer your notes into the relevant folders on a **regular basis**.

Task 2 – Futures in Comp Sci

Watch this video - <https://www.youtube.com/watch?v=IDFAg9oluv8>

Task 3 – Please complete the following questionnaire

You are not expected to have programmed before however it is useful to know what experience (if any) you do have. Please answer all questions you are able to.

1. Which language(s) have you used to write computer programs?

.....

.....

Student Name:

2. For each of the following programming concepts rate your level of experience/expertise in your chosen language.

Concept	No experience	Little experience	Lots of experience
Input from keyboard and Output to screen			
Branching (If..Else or Case statements)			
Iteration – loops. Repeating sections of code several times (e.g. For, While and Repeat)			
Using in-built functions (e.g. maths functions)			
Writing your own functions and procedures (subroutines)			
Using parameters (passing values to functions or procedures)			
Manipulating strings (joining, splitting and searching in one string for another)			
Using Arrays or Lists to store and manipulate collections of values			
Reading from and/or writing to text files			
Working with Records and Fields (possibly with databases)			
Drawing and graphics			
Animation			
Writing classes			

3. In what contexts have you written programs?

Context	Tick all that apply
Exercise or project at school	
For own interest	
Software in use by others	
Other (please state):	

Student Name:

Task 4 – Algorithms

An algorithm sets out a sequence of steps, that, when executed, will carry out a specific task. Please complete the following algorithms questions, which are examples of GCSE examination questions.

Note: We are not trying to see if you can gain full marks, but rather how you approach and whether you enjoy this style of questions. Hopefully, it will help you to confirm that this is the right course for you!

1. A programmer has written an algorithm to output a series of numbers. The algorithm is shown below in OCR Pseudocode:

```
01 for k = 1 to 3
02 for p = 1 to 5
03 print (k + p)
04 next p
05 next k
06 m = 7
07 print m * m
```

- (a) Give the first **three** numbers that will be printed by this algorithm.

.....[1]

- (b) State how many times line **03** will be executed if the algorithm runs through once.

.....[1]

Student Name:

2. During the implementation of an algorithm, a programmer attempts to swap over the value of two variables. The code used is shown below:

```
x = input("enter first number : ")
y = input("enter second number : ")
//swap values over
x = y
y = x
```

- (a) The values **12** and **20** are inputted into this algorithm as the first and second number. Give the values of x and y once this program has been executed.

x

y

[2]

- (b) Complete the program below so that the numbers are successfully swapped over.

```
x = input("enter first number : ")
y = input("enter second number : ")
//swap values over
..... = y
y = .....
x = .....
```

[3]

- a) Tick (3) **one** box per row to show whether each of these constructs are used in the program in part (b).

Construct	Is used in (b)	Is not used in (b)
Sequence		
Selection		
Iteration		

[3]

