

Preliminary Programming 1

Materials Required

Moodle: <http://karabardec.mrooms.net/course/view.php?id=206>

Booklet: Preliminary Programming booklet 1 (on Moodle)

Textbook: Software Design and Development, the Preliminary Course (2nd Edition) by Samuel Davis

Introduction

This lesson you will start working on designing and coding software solutions. It is important that you complete all of the activities in each lesson. Contact your teacher if you have any problems completing the work. You will not use the textbook this lesson but you must make sure that you have a copy or that it has been ordered as it is an essential learning tool for this course.

There is quite a lot to do this week, but remember, you are allocated the equivalent of 6 x 40 periods or 4 hours to complete this subject and you should spend this amount of time completing the work each week.

Submission

Once you have completed the work for each week, save all of your work, which may include answers to questions as well as files and folders that contain the practical activities, in a folder named the same as the Weekly Worksheet number shown on your *Course Program*, for example WW1. ZIP the folder and upload the work for each week into the appropriate *Dropbin*.

You must return all of the files for all programs, both Pascal and Visual Basic. All Pascal Files should be in a folder called Pascal and VB programs should be in separate folders. These folders should be placed in the folder named the same as the Weekly Worksheet along with any other work, such as this document, that you complete for each lesson.

This is explained in more detail in the *Course Instructions* document which you should have already read. Make sure you have read through and completed all of the instructions in the *Course Instructions* document.

Tasks to Complete

- Download and save the *Preliminary Programming 1* booklet to your computer.
- **Read** pp3 – 26 in the *Preliminary Programming 1* booklet, completing all programming examples as described.
- **Complete** Exercises 1 – 6 in the *Preliminary Programming 1* booklet.
- **Submit** the completed exercises in the *Preliminary Programming - Introduction to Software Dropbin*.

Exercise 1

- a. Define and explain the difference between algorithms, pseudocode and flowcharts.

Algorithms are a series of steps used to complete a goal, i.e., if you were to make an algorithm in Python to create a countdown, it would look like this:

```
def countdown(count):  
    while (count >= 0):  
        print ('Hacking complete in: ', count)  
        count -= 1  
        time.sleep(1)  
  
countdown(10)
```

Pseudocode is simplified code that is written out in English, read top to bottom, left to right and has it's important keywords capitalised. An example of Pseudocode can be written out as a joke:

A programmer's wife tells her husband to go to the shop to buy some food. She say's to him "GO to the shop and BUY a loaf of bread, IF they have eggs, BUY a dozen."

She nearly killed him when he returned home with 12 loaves of bread. ☺ Good, but in an exam situation keep the jokes out, but I did enjoy reading it.

Flowcharts are a visual tool used to portray what steps area needed before a project gets on the way. Until pseudocode became so popular, Flowcharts were used as the default pre-project mind map. ☺

Write an algorithm in **pseudocode** to describe how to make a peanut butter sandwich.

REPEAT

INPUT "Do you have peanut butter and Sandwich bread in front of you?"

IF input = "No"

THEN OUTPUT = "Collect peanut butter and Sandwich bread, then place them in front of you."

IF input = "Yes"

THEN OUTPUT = "Place two pieces of bread next to each other, then open the peanut butter jar."

UNTIL input = "Yes."

REPEAT

INPUT "Do you have two pieces of bread in front of you and an open peanut butter jar?"

IF input = "No"

THEN OUTPUT = "Place two pieces of bread next to each other, Open the peanut butter jar and place it near the bread slices."

IF input = "Yes"

THEN OUTPUT = "Spread a knife full of peanut butter onto one slice of bread then place second slice on top of the buttered slice!"

UNTIL input = "Yes"

IF input = "Yes"

THEN OUTPUT = "Enjoy your sandwich"

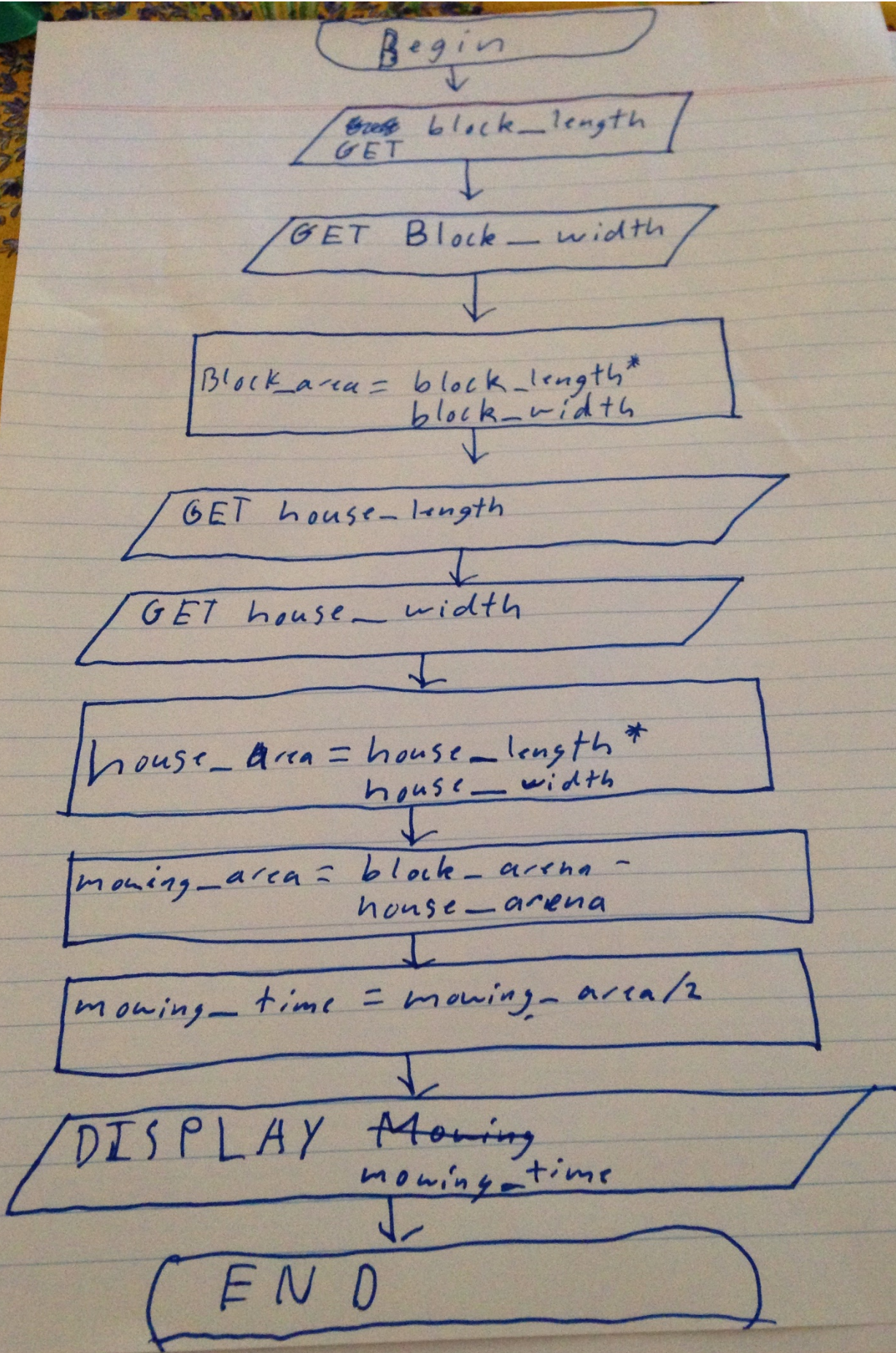
IF INPUT = "No"

THEN OUTPUT "You are beyond saving, just go buy a sandwich!" Good, but it would be better to use GET and DISPLAY here

- b. Draw the **flowchart** for this algorithm. (It may be easier to hand draw it on paper, scan and then copy and paste it into this document) {I couldn't get the hang of it and wasted 2 days trying and failing at making it, so in order to finish the rest of my work I've opted to skips this.}

Exercise 2

Use the problem statement, IPO chart and pseudocode to draw an appropriate flowchart.



Good work here Maanu 😊

Exercise 3

Complete a desk check using the pseudocode from **Exercise 2**. Use the format for desk checking shown in the booklet – use an input table, expected results table and desk check table.

Exercise 4

- a. Define and explain the difference between identifiers, variables and constants.

Identifiers are words that represent numbers or stored data. For example, “PROGRAM” and “Pi”.

Variables are numbers and words (**identifiers**) in code that are changeable or don’t have a consistent value.

Constants (**are identifiers too**) are words and numbers that are consistent and do not change. For example, Pi is always equal to 3.1415 and Sqrt is always $X^{**1/2}$.

- b. What is a syntax error?

Syntax error is what happens when you write incorrect code. For example, {‘Hello world’}; is a syntax error because it uses curly bracket {} not parenthesis (). **Good work**

- c. For each of the following, identify whether they are valid or invalid Pascal identifiers. For those that are invalid, explain why.

- i. birthday

Valid because it holds the numerical data for the birthday 😊

- ii. Too_hot?

Too_hot is invalid because it’s a question not an identifier 😊

- iii. First_Initial

Valid because it holds the First_Initial data. 😊

- iv. grade

valid it would hold the number data for “Grade” 😊

- v. 1stprogram

Valid because it holds the 1stprogram data 😊

- vi. down.to.earth

invalid because it doesn’t hold data, it’s a statement 😊

- vii. see you

it’s a statement so is invalid 😊 **as it has a space inbetween**

- viii. OldName

Valid because it holds the data for OldName 😊

- ix. Begin

Valid because it holds the data for the command BEGIN, used to begin the program **This is invalid Maanu as it is reserved.**

- d. Explain the difference between WRITE and WRITELN in Pascal.

WRITELN puts down what’s written, whereas WRITE allows you to tell it to write in variables as we get them, this is used very often in game. For example, WRITE (a = 21, ‘); would display a = 21, a where WRITELN (‘a = ',a) displays a = 21 **The difference is that WRITELN places return at the end of it, and WRITE doesn’t.**

- e. Write a PASCAL program that displays your name and school on separate lines. Include comments. Name your program *MyDetails*. Place the program files (.pas and .exe) in a folder called PASCAL and place this folder inside the WW folder for submission with this week's work.
- f. Type and run the example programs, *Sounds*, *Colour*, *Screen_coordinates*, and *Clear_Display*, to see how they work **I hope that you have done this Maanu. If you haven't please get this done when you are able.**

Exercise 5

- a. Type and run the example programs *Variables* and *ReadDemo* to see how they work.
- b. Use the *Preliminary Programming 1* booklet to find the value of the following expressions. Your answers must be typed and included in your returned work. Show all working.
 - i. $27 - 23 \text{ div } 5$
22.4 (23 div 5 = 4 and so 27 - 4 = 23)
 - ii. $(27 - 23) \text{ div } 5$
0.8 so this results in 0.
 - iii. $2 * (45 \text{ mod } 7) - 4$
2 😊
 - iv. $\text{abs}(-9) * \text{sqr}(3)$
15.6 = 9 * 9 = 81
 - v. $\text{sqrt}(\text{trunc}(9.143))$
 - vi. $10 * \text{round}(0.876)$
- c. Use the **Reference – Free Pascal** section at the end of *Preliminary Programming 1* booklet to complete the following:
 - i. What colour do reserved identifiers change to in Free Pascal
 - ii. What does the APPEND keyword do?
 - iii. When would you use the CHR(arg) function?
 - iv. What predefined procedure would you use to put the cursor in a specific location on the screen?
- d. Construct an algorithm and then a Pascal program that asks a user for their name and age and then prints their name and age on separate lines. Name your program *UserDetails*. Place your completed program (.pas and .exe) into the PASCAL folder you created earlier.
NOTE: if you place READLN as the last statement in your program it will allow you to view the solution before the program ends. Occasionally you may need to add READLN twice, on separate lines.
- e. Construct an algorithm and then a *Pascal* program (*GrossPay.pas*) to calculate the gross pay. Ask the user for the number of hours worked, the rate of pay is \$11.45 per hour. Rate should be a constant, the output must be formatted to two decimal places and include a \$ in the answer. Gross_pay is calculated by multiplying hours_worked by rate. Place your completed program (.pas and .exe) into the PASCAL folder you created earlier.

- f. Create a program (*FourNumbers*) that gets the user's input (four numbers) and finds a solution using the following formula. You must use the *Pascal* arithmetic symbols shown in the booklet. Place your completed program (.pas and .exe) in the PASCAL folder you created earlier.

$$\text{answer} = \frac{\text{num1} \times \text{num2} + \text{num3} \times \text{num2}}{\text{num4}}$$

Exercise 6

- Complete and submit *MyFirstApp* using *Visual Basic Express 2008*.
- Complete and submit *ErrorTrapping* using *Visual Basic Express 2008*.

Make sure that you place **all of the files and folders** for each VB program into the WW folder before compressing and uploading the folder to Moodle.

- Use the **Reference – Visual Basic 2008** section at the end of the *Preliminary Programming 1* booklet to answer the following:
 - Use the proper conventions to write the name for an identifier called School that is a string data type.
 - Write down the name, using the proper conventions, for a timer object called Clock.
 - What is meant by the *scope* of an identifier?

Assessment Task 1

- Acknowledge that you have completed the *Receipt of Course Materials* Feedback Activity on Moodle.

I did it.

Open AT1 (RAD project) and complete the following tasks:

- Read the all of the task including the Marking Criteria and note down at least one question that you have about the task.

You can of course, ask as many questions as you would like. You are encouraged to ask questions and contact your teacher when you need help or if you are unsure about what you must do.