Name: Class:

**Task 1**

1. Programming languages are sometimes categorised by ‘generations’. Complete the table below:

|  |  |  |
| --- | --- | --- |
| **Generation** | **Name of language** | **Example instruction** |
| 1st generation |  |  |
| 2nd generation | Assembly language |  |
| 3rd generation |  |  |

2. Match each of the following terms with their meaning.

|  |  |  |
| --- | --- | --- |
| **Term** |  | **Meaning** |
| Machine code |  | The tool that translates assembly code into machine code. |
| High-level language | The binary code that will be executed when a program is run. |
| Low-level language | A language that is similar to English and easier for programmers to read and write. It has more advanced structures such as FOR loops. |
| Operand | The specific operation of an instruction. This will be stored as a binary number, but in assembly it is represented by a mnemonic such as MOV for ‘move’ or JMP for ‘jump’. |
| Operator | The values or memory addresses that an opcode will work on. |
| Instruction set | All the possible instructions that are available on a particular CPU. |
| Assembler | A language that is able to program precisely what the CPU or hardware will do – the language used is known as assembly language |

Task 2

1. A programmer is attempting to run a simple program. After writing the code they press the “Run” button to run the program. The following screen is immediately displayed:



Decide whether the program has been compiled or interpreted, then explain your decision.

2. In the programming language Python, you can program in either of two different modes. One mode allows programs to be developed and run; the interactive mode allows the program to be entered using a text interface.



 Are the statements in this program compiled or interpreted? Describe what is happening in the above screenshot.

3. Most programs are made by creating the program code in a text editor and saving it. The program will then either be compiled before it is executed, or it will be interpreted.

 Look at the following section of code:

total = 0

for i = 1 to 100

 total = total + i

print("Total = ", total)

 How many lines of code would be translated using -

(a) an interpreter?

(b) a compiler?