Create a project folder named “lab\_test” on your desktop. All your lab test answers should be saved in this folder. Create a new project called “lab\_test\_project”, including all settings and configurations, then key in the Q1 program as given below. Name the Q1 program as “Q1\_solution.c”. Compile and run the program. Signal your invigilator to verify the answer and sign on the lab test sheet.

|  |
| --- |
| **#include <f200.h>** |
| **void setSystem();** |
| **void main()** |
| **{** |
| **setSystem();** |
| **P1 = 0xF0;** |
| **for(;;)**  {  P1 = 0x00;  delay(10000);  P1 = 0x01;  delay(10000);  } |
| } |

There are **5** syntax errors and **5** logical errors in the program given below. Correct all errors on the paper and indicate whether it is Syntax or Logical error. Key in your program as Q2\_solution.c, compile and run. Signal invigilator to verify and sign.

|  |  |  |
| --- | --- | --- |
| **Code** | **Correction** | **Synt/Log** |
| #include <f200.h> |  |  |
| void delay(unsigned long x) |  |  |
| { |  |  |
| while(x==0); | while(x--!=0) | Log |
| } |  |  |
| unsigned char keyscan(unsigned char \*z) |  |  |
| { |  |  |
| if else (P34==1) | if | Synt |
| { |  |  |
| \*z = P3 & 0xF0; | \*z = P3 & 0x0F; | Log |
| return(1) | return(1); | Synt |
| } |  |  |
| else |  |  |
| { |  |  |
| return(0); |  |  |
| } |  |  |
| } |  |  |
| void setSystem(); |  |  |
| void main() |  |  |
| { |  |  |
| unsigned char Key; | unsigned char key; | Synt |
|  | setSystem(); | Log |
| P1 = 0x00; |  |  |
| for(;;) |  |  |
| { |  |  |
| if(keyscan(&key) ==0) | if(keyscan(&key) ==1) | Log |
| { |  |  |
| switch(key) |  |  |
| { |  |  |
| case 1 | case 1: | Synt |
| P1=0x01; break; |  |  |
| case 2: |  |  |
| P1=0x03; | Break; | Log |
| default: P1=0x00; break; |  |  |
| } |  |  |
| } |  |  |
| } |  |  |
|  | } | Synt |

|  |  |  |  |
| --- | --- | --- | --- |
| **Assessment** | **Signature** | **Marks** | **Comments** |
| Program can be compiled without error and runs successfully. (10%) |  |  |  |

Write ONE program and save it as Q3\_solution.c to perform the following operations. Signal to your invigilator to verify your result. If you are unable to show a running program, write the main( ) section and user defined global variable(s), if any, in the space provided. There is no need to write any of the function or routine used in the lab.

1. When program is first executed, display the following pattern on the LED bar:
2. With each press of SW2 (without holding), cycle through one step at a time the following LED patterns on the LED bar with each press of SW1:

….

….

1. Meanwhile if SW2 is pressed any time, reset to the initial LED pattern on the LED bar: