**Part 1: Use notebook**

**Exercise 1** (1 mark) : Explain outputs:



\*pn = n = 7; \*pm = m = 6

\*pn = \*pm + 2\*m -3\*n = 3(m -n) = 3(6-7) = -3

\*pm = \*pm - \*pn = m – n = 9

🡺m+n= -3 + 9 = 6



\*p1 = c1 = ‘A’; \*p2 = c2 = ‘F’

\*p1 = \*p1 + 3 = ‘A’+3

\*p2 = \*p2- 5 = ‘F’ -5

🡺\*c1-c2 = \*p1 - \*p2 = ‘A’ + 3 – ‘F’ + 5 = 3



\*p1 = x = 3.2; \*p2 = y = 5.1

\*p1= \*p1 + 3 – 2(\*p2) = 3.2 + 3 – 2(5,1) = 6,2 – 10.2 = -4

\*p2 = \*p2 – 3(\*p1) = 5,1 – 3(-4) = 17,1

🡺x+y = \*p1+\*p2 = 17,1 – 4 = 13.1

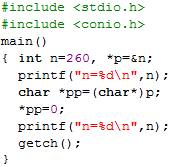
**Exercise 2: (1 marks) What are outputs**

n = \*p1 =7; m = \*p2 = 8;

\*p1= n = 12- m +\*p2 = 12 -7 +8 = 13

\*p2 = m + n -2\*(\*p1) = 8 + 13 – 2\*n = -5

m+n = 13-5 = 8

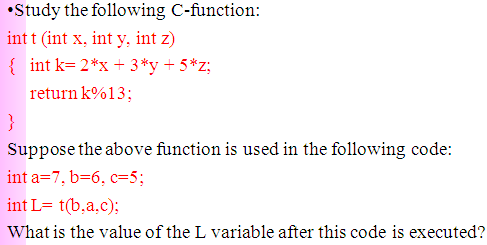


n + \*p = 260;

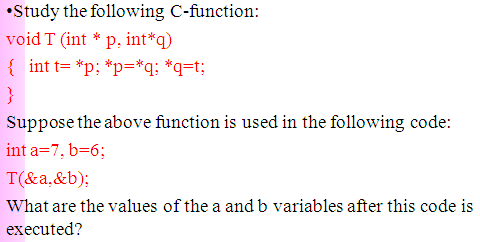
🡺260

🡺0

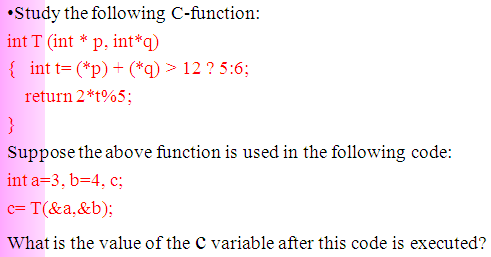
**Exercise 3: (2 marks) Walkthroughs**



a =7, b=6, c=5 t(b,a,c) ⌠ x = 6, y = 7, z = 5 k = 2\*x+3\*y+5\*z = 2\*6 + 3\*7 + 5\*5 = 58 return k%13 => t(b,a,c) = 6.



a = 7, b = 6 T(&a,&b) ⌠ \*p = &a \*q = &b t = \*p, \*p = \*q; \*q = t;  a = 6, b =7.



a = 3, b = 4, c T=(&a,&b) ⌠ \*p = 3, \*q = 4 t = (\*p) + (\*q) > 12?5:6 = 3 + 4 < 12 False => t = 6 Return2\*t%5 => T(&a,&b) = 2.

**Part 2: Develop a program using simple menu**

**Program 1(3 marks):**

|  |  |
| --- | --- |
| **Objectives** | Practice implementing a program with simple menu. |
| **Related knowledge** | None |
| **Problem** | Write a C program that will execute repetitively using a simple menu as following:   1. **Process primes** 2. **Print min, max digit in an integer;** 3. **Quit**   **Select an operation:**   1. When user selects the option 1, the program will accept a positive integral number and print out a message about whether the input number is a prime or not. 2. When user selects the option 2, the program will accept a positive integral number and print out the minimum and maximum digit in this number. 3. The program will terminate when user selects the option 3. |
| **Analysis** | **Nouns:**  - positive integral number 🡪 **int n**  - A number represents a choice of user 🡪 **int choice;**  **Functions**:  **int prime( int n) 🡪 see above**  **void printMinMaxDigits( int n) 🡪 see above** |
| **Suggested algorithm (logical order of verbs)** | Begin  Do /\* Print out the menu and get user choice\*/  { Print out “1- Process primes\n”;  Print out “2- Print min, max digit in an integer \n”;  Print out “3- Quit\n”;  Print out “Select an operation:”;  switch(choice)  { case 1: do  { Input n;  }  while(n<0);  If ( prime(n)==1) Print “ It is a prime\n”;  Else Print “ It is not a prime\n”;  break;  case 2: do  { Input n;  }  while(n<0);  printMinMaxDigits( int n) ;  break;  }  }  while ( choice >0 & choice<3);  End |

**Program 2(3 marks): ( refer to the workshop 2 for algorithms)**

Write a C program that will execute repetitively using a simple menu as following:

**1-Fibonacci sequence**

**2-Check a date**

**3-Quit**

**Choose an operation:**

1- When the option 1 is selected, the program will accept a positive integral number, called as n, then the first n Fibonacci numbers will be printed out

2- When the option 2 is selected, the program will accept a date then the program will tell that whether this data is valid or not.

3- If the option 3 is selected, the program quits

**More Programs**

You can pick 2 or 3 functions in the workshop 2, associate them to a new program.