**Subject: PRF192- PFC**

**Workshop 04**

**Tên: Trần Nguyễn Quốc Cường**

Tiếp thu 90%, điểm slot: 8đ; ý kiến:…..

**Objectives:**

1. Managing data using pointers
2. Developing programs using simple menus

**Part 1: Use notebook**

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



Program 1:

n = 7, m = 6

\*pn = \*pm +2\*m-3\*n 🡺 n = m+2\*m-3\*n

n = -9, m = 6

\*pm -=\*pn 🡺 m = m – n

n =-9, m = 15

m + n = 6

Program 2:

c1 = ‘A’, c2 = ‘F’

\*p1 += 3 🡺 c1 = c1 + 3

c1 = ‘D’, c2 = ‘F’

\*p2 -= 5 🡺 c2 = c2 – 5

c1 = ‘D’, c2 = ‘A’

c1 – c2 = 3

Program 3:

x = 3.2, y = 5.1

\*p1 += 3 – 2\*(\*p2) 🡺 x= x + 3 – 2\*y

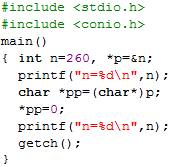
x = -4, y = 5.1

\*p2 -= 3 \* (\*p1) 🡺 y = y – 3 \* x

x = -4, y = 17.1

x + y = 13.1

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



Program 1:

n = 7, m = 8

n += 12 – 8 + 8 🡺 n = 7 + 12 = 19

n = 19, m = 8

m = 8 + 19 – 2 \* 19 = -11

n = 19, m = -11

n + m = 8

So output is 8

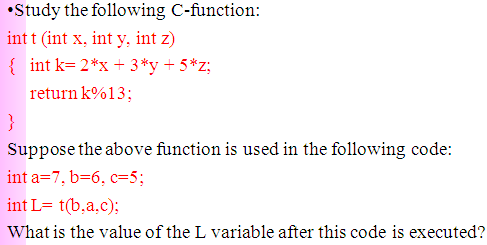
Program 2:

n = 260(base 10) 🡺 1 0000 0100(base 2)

\*pp = 0 🡺 4 lower bits turn to 0 🡺 n = 1 0000 0000(base 2) 🡺 256(base 10)

So output is 256

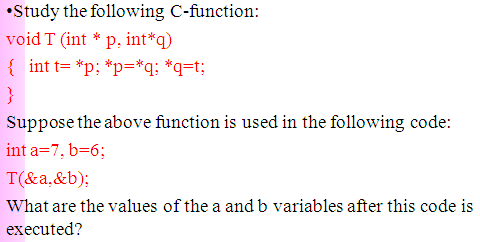
**Exercise 3: (2 marks) Walkthroughs**



k = 2 \* 6 + 3 \* 7 + 5 \* 5 = 58

L = 58 % 13 = 6

So L = 6

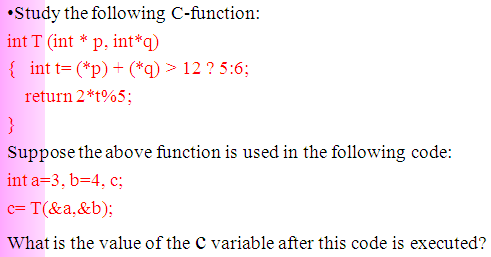


t = 7

a = 6

b = 7

So a = 6, b = 7



a + b = 3 + 4 = 7 < 12 🡺 t = 6

c = 2 \* 6 % 5 = 2

So c = 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 |

/\*

Name: Tran Nguyen Quoc Cuong

Date: 2022/02/21

\*/

#include <stdio.h>

#include <math.h>

int prime(int n) {

int result = 1;

float m = sqrt(n);

for (int i = 2; i <= m; i++) {

if (n % i == 0) result = 0;

}

return result;

}

void printMinMaxDigits(int n) {

int digit;

int min = 9;

int max = 0;

while (n > 0) {

digit = n % 10;

if(digit > max) max = digit;

if(digit < min) min = digit;

n = n / 10;

}

printf("Min digit is: %d\n", min);

printf("Max digit is: %d\n", max);

}

int main() {

int c, n;

do {

printf("1 - Process primes\n");

printf("2 - Print min, max digit in an integer\n");

printf("3 - Quit\n");

printf("Select an operation: ");

scanf("%d", &c);

switch(c) {

case 1:

{

do {

printf("Input n: ");

scanf("%d", &n);

} while (n < 0);

if (prime(n) == 1) printf("%d is a prime\n", n);

else printf("%d is not a prime\n", n);

break;

}

case 2:

{

do {

printf("Input n: ");

scanf("%d", &n);

} while (n < 0);

printMinMaxDigits(n);

break;

}

}

} while ((c > 0) && (c < 3));

return 0;

}

**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

/\*

Name: Tran Nguyen Quoc Cuong

Date: 2022/02/21

\*/

#include <stdio.h>

void printFibo(int n) {

int a = 1, b = 1, f = 1;

if (n == 1) printf("1");

else if (n == 2) printf("1 1");

else {

printf("1 1 ");

for (int i = 3; i <= n; i++) {

f = a + b;

a = b;

b = f;

printf("%d ", f);

}

}

}

int validDate (int d, int m, int y) {

int maxd = 31;

if (d < 1 || d > 31 || m < 1 || m > 12) maxd = 0;

if (m == 4 || m == 6 || m == 9 || m == 11) maxd = 30;

else if (m == 2) {

if ((y % 400 == 0) || ((y % 4 == 0) && (y % 100 != 0))) maxd = 29;

else maxd = 28;

}

return maxd;

}

int main() {

int c, n, d, m, y;

do {

printf("1 - Fibonacci sequence\n");

printf("2 - Check a date\n");

printf("3 - Quit\n");

printf("Select an operation: ");

scanf("%d", &c);

switch(c) {

case 1: {

do {

printf("Input n: ");

scanf("%d", &n);

} while (n < 1);

printFibo(n);

printf("\n");

break;

}

case 2: {

int d, m, y;

printf("Input date: ");

scanf("%d", &d);

printf("Input month: ");

scanf("%d", &m);

printf("Input year: ");

scanf("%d", &y);

if (validDate(d, m, y) >= d) printf("Valid date\n");

else printf("Invalid date\n");

break;

}

}

} while ((c > 0) && (c < 3));

return 0;

}

**More Programs**

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

/\*

Name: Tran Nguyen Quoc Cuong

Date: 2022/02/21

\*/

#include <stdio.h>

int continuousSum(int n) {

int S = 0;

do {

printf("Input n: ");

scanf("%d", &n);

S = S + n;

} while (n != 0);

return S;

}

void continuousSwap(int x, int y) {

do {

printf("Input x: ");

scanf("%d", &x);

printf("Input y: ");

scanf("%d", &y);

int c = x;

x = y;

y = c;

printf("New x: %d\n", x);

printf("New y: %d\n", y);

} while (x != 0 && y != 0);

}

int main() {

int c, n, x, y;

do {

printf("1 - Continuous sum, stop when n = 0\n");

printf("2 - Continuous swap, stop when x = 0 and y = 0\n");

printf("3 - Quit\n");

printf("Select an operation: ");

scanf("%d", &c);

switch(c) {

case 1: {

printf("Sum of inputted numbers is: %d\n", continuousSum(n));

break;

}

case 2: {

continuousSwap(x, y);

break;

}

}

} while ((c > 0) && (c < 3));

return 0;

}