Subject: PRF192- PFC

Workshop 02

Objectives:

(1) Practicing skills at analyzing and implementing programs using user-defined functions.

(2) Making familiar with some basic algorithms

Grading 4 programs, 2.5 mark/program

Program 1:

Objectives	Practice implementing simple functions
Related knowledge	Fibonacci sequence: 1 1 2 3 5 8 13 21 34 … Two first numbers: 1 Others: Its value is the sum of 2 previous numbers
Problem	Write a C program that will print out the value at the n^{th} position in Fibonacci sequence.

Analysis
A position
➔ int n
	Suggested algorithm (logical order of verbs)
	Begin
	Do {
	Accept n;
	}
	While (n<1);
	Print out fibo(n);
	End.

Algorithm for Computing the n^{th} value of the Fibonacci sequence	**double fibo (int n) {** **int t1=1, t2=1, f=1, i ;** **for (i= 3, i<=n; i++) {** **f= t1 + t2;** **t1= t2;** **t2=f;** **}** **return f;** **}**

How to compute the nth value of the Fibonacci sequence

Position 1	2	3	4	5	6	7	8	9	10
1	1	2	3	5	8	13	21	34	55
T1	T2	F							
	T1	T2	F						
		T1	T2	F					
			T1	T2	F				
				T1	T2	F			
					T1	T2	F		
						T1	T2	F	…

Program 2:

Objectives	Practice implementing simple functions
Related knowledge	
Problem	Write a C program that will accept a positive integer then print out whether it is an element of the Fibonacci sequence or not.

Analysis
An integer → int n
	Suggested algorithm (logical order of verbs)
	Begin
	Do {
	Accept n;
	}
	While (n<1);
	If (isFibonacci(n)==1) Print out "It is a Fibonacci element.";
	Else print out "It is not a Fibonacci element."
	End

Algorithm for Checking whether an integer is a element of the Fibonacci sequence or not	**int isFibonacci (int n)**
	{ int t1=1, t2=1, f=1;
	if (n==1) return 1; /* n belongs to the Fibonacci sequence*/
	while (f<n) /* Find out the Fibo number f to n */
	{ f= t1 + t2;
	t1=t2;
	t2=f;
	}
	return n==f; /* if n==f → n is Fibo element → return 1 */
	}

Program 3:

Objectives	Practice implementing simple functions
Related knowledge	Getting the rightmost digit of the integer n: **n%10**
Problem	Write a C program that will carry out some times. In each time, a nonnegative integer is accepted then print out the sum of its decimal digits. The program will terminate when its value of accepted number is negative.

Analysis
Sum → int S=0
Accepted integer
→ int n
	Suggested algorithm (logical order of verbs)
	Begin
	Do
	{ Accept n;
	If (n>=0)
	{ S = sumDigits(n);
	Print out S;
	}
	}
	While (n>=0);
	End

Algorithm for Computing sum of digits of a nonnegative integer	**int sumDigits (int n)** **{** int sum=0; /* initialize sum of digits */ Do { int remainder = n%10 ; /* Get a digit at unit position */ n = n/10;

	sum += remainder;
	}
	while (n>0);
	return sum;
	}

Program 4:

Objectives	Practice implementing simple functions

Related knowledge	Find out the greatest common divisor **(gcd)** and least common multiple **(lcm)** of two positive integers:
	Find out gcd of a and b
	a b a b
	14 21 13 8
	14 7 5 8
	7 7 5 3
	2 3
	2 1
	1 1
	int gcd(int a, int b)
	{ while (a != b)
	if a>b then a -=b;
	else b -= a;
	return a;
	}
	int lcm (int a, int b)
	{ return a*b/ gcd(a,b);
	}

Problem	Write a C program that will accept two positive integers then print out their greatest common divisor and least common multiple.

Analysis
Two integers
➔ int a, b
gcd → int d
lcm → int m
	Suggested algorithm (logical order of verbs)
	Begin
	Do
	{ Accept a, b;
	}
	While (a<=0 OR b <=0);
	d = gcd(a,b);
	m = lcm (a.b);
	Print out d;
	Print out m;
	End