


2009.06.23

**NSS ICT programming module
from CIT+ALCS to ICT**

CHUNG Wai Tung



Background

- 2009-2012 First NSS ICT implementation
- **CIT Module A + ALCS Paper 2 = ICT Module D** (*Software Development*)
- Main focus of ICT Module D
 - (1) programming, **algorithms and data structure** 47 hours (CIT level + ALCS level / 2)
 - (2) programming languages and **translator technologies** 12 hours (ALCS level)
 - (3) system development - **software development cycles** 16 hours (ALCS level)

Major **objectives** of teaching programming in HK secondary schools

- ICT exam
- Write real programs
- Join HKOI / CCC / NOI / IOI / ACM programming competitions
- Bridging for U-level IT-related subjects professional training

Quick scan – **programming I**

- Top-down, bottom-up, stepwise refinement, modularity, structured programming, structured data types, user-defined data types, **set**, flowcharts, block diagrams, counting, accumulating, swapping, searching, sorting, merging, linear search, binary search, bubble sort, insertion sort, **merge sort**, merge two arrays, **realize quick sort**

old ALCS syllabus only more attention

Quick scan – **programming II**

- **Complexities**, data structures, efficiency, correctness, appropriateness, global variables, local variables, parameters passing, **call by value**, **call by reference**, precedence and association, sequence, selection, iteration, **recursion**, lists, stacks, queues, **linear linked lists** in terms of arrays, **binary tree**, text file **updating**, syntax/logical/runtime errors, rounding/truncation/overflow/underflow errors, stubs, flags, break point

old ALCS syllabus only more attention

Quick scan – **programming languages**

- Procedural, logic, object-oriented, query languages, selection criteria, compilers, interpreters, code generation, linkers, loaders, **lexical analyzer**, **parser**, **semantic analysis**, **syntactic analysis**, **symbol table**, **token strings**, **parse trees**, **object program**

old ALCS syllabus only more attention

Quick scan – system developments I

- Environment, inputs, outputs, processes, interfaces, storage, **Waterfall Model**, gather information, interviews, surveys, questionnaires, observations, document review, **users' requirements, feasibility study, proposed solution, PERT chart, Data Flow Diagram, test plan**

old ALCS syllabus only more attention

Quick scan – system developments II

- System **conversion**, pilot / phased / parallel / direct cutover, **ongoing maintenance**, upgrading, training, system documentation, **technical documentation**, user manuals, requirement statements, project plans, design plans, **prototyping, RAD, job titles**

old ALCS syllabus only more attention

Choose an appropriate algorithm for a task : searching, merging, sorting, recursion, etc.

- A kind of application skill
- Task-oriented
- Problem-based learning

ID	Problem name
1000	HelloWorld
1001	A+B Problem
1002	Quirksome square
1003	Hangman Judge
1004	Manok Mark
1005	Bridge
1007	Bitmap - Hexagon
1008	The Lottery
1009	Magic Squares
1010	Sum of multiples
1011	Prime factors

Time complexity – Big-O

Notation	Name	Example
$O(1)$	constant	Determining if a number is even or odd
$O(\alpha(n))$	inverse Ackermann	Amortized time per operation when using a disjoint-set (union-find) data structure
$O(\log^k n)$	iterated logarithmic	The ϵ -BFS algorithm of Hopcroft and Ullman on a disjoint set
$O(\log n)$	logarithmic	Finding an item in a sorted list with the binary search algorithm
$O((\log n)^c)$	polylogarithmic	Deciding if n is prime with the AKS primality test
$O(n^c), 0 < c < 1$	fractional power	searching in a kd-tree
$O(n)$	linear	Finding an item in an unsorted list
$O(n \log n)$	linearithmic, loglinear, or quasilinear	Sorting a list with heapsort, computing a FFT
$O(n^2)$	quadratic	Sorting a list with insertion sort, computing a DFT
$O(n^c), c > 1$	polynomial, sometimes called algebraic	Finding the shortest path on a weighted digraph with the Floyd-Warshall algorithm
$O(c^n)$	exponential, sometimes called geometric	Finding the (exact) solution to the traveling salesman problem (under the assumption that $P \neq NP$)
$O(n!)$	factorial, sometimes called combinatorial	Determining if two logical statements are equivalent [1]#, traveling salesman problem, or any other NP-complete problem via brute-force search

Implement parameters passing in manipulating sub-programs

```

chungsir
隨時使命
加入日期: 2003-05-11
文章數: 1504
科目: PCC1-藍球院
P.119 Question 1.
(a) key : no parameter, all are global variables
Answer : 1 3 4
(b) key : C is local variable (cannot affect global C)
Answer : 1 3 5
(c) key : X, Y, Z are value parameters (never return to the main)
Answer : 1 3 5
(d) Key : X and Y are variable parameters, Z is value parameter
Answer : 10 20 5
MiniMini Quiz - modify P.120 (1) (d)
program parameters_passing;
var A, B, C : integer;
procedure FOUR(var X, Y : integer; Z : integer); (* <- Param
begin
X := 10; Y := 20; Z := 30;
end;
begin (* main program *)
A := 1; B := 3; C := 5;
FOUR(A, B, C);
writeln(A, B, C);
end.
(e) Change Four(A, B, C) -> Four(C, B, A);
(f) Change Four(A, B, C) -> Four(A, C, B);
(g) Change Four(A, B, C) -> Four(A, A, A);
(h) Change Four(A, B, C) -> Four(B, A, B);
  
```

(d) Answer
10 20 5
(e) Answer
1 20 10
(f) Answer
10 3 20
(g) Answer
20 3 5
(h) error : variable expected

Choose a programming language

Five main types of computer programming languages (references/wikipedia)

- 1) Declarative programming languages
e.g. Lisp [1956], Prolog (1970), SQL (1970)
- 2) Procedural programming languages
e.g. Fortran (1954), COBOL (1959), BASIC (1964), Pascal (1970), C (1971), Ada (1974), C++ (1983), Visual Basic (1991), PHP (1994), Java (1994)
- 3) Object-oriented programming languages
e.g. Simula (1962), C++ (1983), Java (1994)
OO features have been added to many existing languages during that time, including Ada, BASIC, Lisp, Fortran, Pascal, Visual Basic, etc.
- 4) Visual programming languages
e.g. Spreadsheet (1983)
- 5) Query languages
e.g. SQL (1970)

Programming language history and the 'shooting yourself in the foot' 🐼

[http://210.176.23.163/course/computer/public/html/\(AL\)%20Program%20Tracing%20\(Prolog,%20C++,%20SQL\)/](http://210.176.23.163/course/computer/public/html/(AL)%20Program%20Tracing%20(Prolog,%20C++,%20SQL)/)

program debugging

TP70之寶：TURBO DEBUGGER

- 大家若要處理比較深奧的問題，必須要開始使用 debugger。TP70 最好用的地方，是它有一個非常 User friendly 的 debugger。

- CTRL-F7 ADD WATCH (加入被監視數據)
- F8 STEP-BY-STEP (一句一句執行)
- CTRL-F2 RESET EXECUTION (由頭再行過)
- ALT-W, TILE (把所有視窗同時顯示)
- F6 SWITCH BETWEEN WINDOWS (換到其他視窗)
- CTRL-break (用戶提出中止執行)

```

begin
  ch := 'b';
  writeln(ch);
  KK;
  writeln(ch);
end;

procedure CC;
var
  ch: char;
begin
  ch := 'c';
  writeln(ch);
  KK;
  writeln(ch);
end;

procedure DD;
var
  ch: char;
begin
  
```

Watches = []
ch: 'c'

Case studies : PCOI and ICT programming

- Automatic program judge website
 - (1) school-level
 - (2) interschool-level
- Everyone can contribute
- Study **Logs** for further references
- Problem-based **24x7** self learning

[Home | task | Submit | Status | User list | User info | Statistic | Logout (wtchung)]

Task

Sorted By AC

Search: []

ID	Problem name	Start date	Time limit	Status	Submit	SOL
1000	helloworld	2006-07-26	1 s	289	Go	1000
1030	Fibonacci sequence	2007-04-21	1 s	163	Go	1030
1009	Magic Square	2006-08-04	1 s	161	Go	1009
1017	Area and Perimeter	2006-08-07	1 s	159	Go	1017
1011	Prime factors	2006-08-06	10 s	150	Go	1011
1085	Pascal triangle	2007-05-08	1 s	150	Go	1085
1024	Climbing Worm	2006-08-04	1 s	142	Go	1024
1026	3 Cups	2006-08-07	1 s	136	Go	1026
1018	Pass or Fail	2006-08-07	1 s	133	Go	1018
1007	Bitmap - Hexagon	2006-08-04	1 s	125	Go	1007
1071	Serial numbers	2006-12-08	1 s	118	Go	1071

Beginner's level

1025	The God of Moon	2006-12-14	1 s	10	Go	1025
1076	Wireless Communication	2006-12-10	1 s	10	Go	1076
1102	Sorting game	2008-10-20	1 s	10	Go	1102
1036	Paper-Scissors-Rock	2006-08-28	1 s	9	Go	1036
1048	Course Allocations	2006-10-22	1 s	8	Go	1048
1073	CIT exam uses Procedures ...	2006-12-09	1 s	8	Go	1073
1082	Nonomino Sudoku	2007-02-12	1 s	8	Go	1082
1072	Binary search tree and its depth	2006-12-08	1 s	7	Go	1072
1078	High Table Dinner	2006-12-12	1 s	7	Go	1078
1047	Number Line	2006-12-12	1 s	7	Go	1047
1083	Save your grandma	2006-12-08	1 s	7	Go	1083
1064	Carrying Laser Cannons	2006-12-01	5 s	5	Go	1064
1062	Wall game	2006-12-01	1 s	4	Go	1062
1069	Marathon	2006-12-05	1 s	2	Go	1069
1034	find -st /	2006-08-25	1 s	1	Go	1034
1070	Simple Matrix Calculation	2006-12-05	3 s	1	Go	1070

Advanced level

Problem status - Wall game

Problem ID	1062
Problem name	Wall game
Solved	4
Accepted	4
Prestation error	0
Wrong answer	45
Runtime error	0
Time limit exceeded	0
Output limit exceeded	0
Compile error	2
System error	0

Solved user list

hkhoo gary those mhon

Status

Problem ID	User ID	Submit time	Language	Status
1062	pyso	2008-07-15 01:32:30	Pascal	Wrong Answer
1062	pyso	2008-07-15 01:23:14	Pascal	Wrong Answer
1062	pyso	2008-07-14 16:52:00	Pascal	Wrong Answer
1062	pyso	2008-07-14 16:50:29	Pascal	Wrong Answer
1062	pyso	2008-07-14 16:49:58	Pascal	Compile Error
1062	pyso	2008-07-14 16:37:45	Pascal	Wrong Answer
1062	pyso	2008-07-14 16:32:48	Pascal	Wrong Answer
1062	pyso	2008-07-14 16:06:19	Pascal	Wrong Answer
1062	mhon	2008-07-14 16:02:40	C++	Accepted

[Home | Task | Submit | Status | User list | User info | Statistic | Logout (wtchung)]

Problem 1062 user mhon

```

#include <stdio.h>
int main(void) {
  char a[100][100];
  bool e[100][100], f;
  int w, h;
  scanf("%d%d", &h, &w);
  for (int i=0; i<h; i++)
    scanf("%s", a[i]);
  if (a[0][0]!='C') e=true;
  if (a[h-1][w-1]!='C') e=false;
  for (int i=0; i<h; i++)
    if (a[i][0]!='W') h=i;
  int j=h;
  for (int i=0; i<w; i++)
    while (a[i][j]!='W') j++;
  if (a[i][j]!='W') h=j;
  else if (a[i][j-1]!='W') h=j-1;
  else h=j;
  if (j-h) f=false;
  else if (h-j) f=true;
  if (e==h&&f)
    if (e==f) h++;
  }
  else
    h--;
  if (e&&f&&h) h++;
  else if (e&&f&&h) h--;
  else h++;
  printf("%d\n", h);
  return 0;
}
  
```

When you have submitted a correct solution, you can check for others answer!

ALCS student

User ID	Solved	User ID	Solved	User ID	Solved
whlaw	62	kyto	26	kin	19
Koa	57	lawrencelee	26	Thomas	19
PedroFok	43	klman	24	winterdesire	15
hefong	38	tohobohu	22	yp_wong	13
htiam	34	OhSez	22	20021125	12
shingingtar	33	chchoi	20	Louis	12
utchan	33	Fung	20	Elaine	6
woy	32	kktyhk	19		
wave	30	tanpepo	19		

Trainer

User ID	Solved	User ID	Solved	User ID	Solved
gary	60	hkho	17	mkleung	0
those	58	billynyh	8	genya	0
ioeki	56	billy	2		
sunny	33	pc-mhh	1		
wchung	25	macau	1		

Macau PCMS

User ID	Solved	User ID	Solved	User ID	Solved
ryan_oda_pcms	21	002_mo	2	005_mo	1
natalie_pcms	12	007_mo	2	lnam_mo	1
kellapcms	10	mak_pcms	2	ohin_mo	1
aypcms	9	harrypcms	2	adapcms	1
lwpccms	7	006_mo	2	athenapcms	1
stevenpcms	6	010_mo	2	008_mo	1

Just do it!

chungsir
隨時解命
加入日期: 2003-05-11
文筆數: 1504
來自: PCOI-籃球隊

發表時間: 2008-10-12 10:07

學習 PROGRAMMING (Module A) 的真正王道是:

Selection

- [Classwork] Exercise 4-1 Nature of Roots
- [Classwork] Exercise 4-2 Right Triangle
- [Classwork] Exercise 4-3 Grade Conversion
- [Homework] Exercise 4-4 Find Roots
- [Homework] Exercise 4-5 Tax (task 1029)

Iteration

- [Classwork] Exercise 5-1 Factorial
- [Classwork] Exercise 5-2 Difference (task 1097)
- [Homework] Exercise 5-3 Integer Series
- [Homework] Exercise 5-4 Compound Interest (task 1028)
- [Homework] Exercise 5-5 Grid (task 1015)

>>>> More and more challenges :

- [Challenge] Exercise 4-6 Prime Factors (task 1011)
- [Challenge] Exercise 4-7 Days Beyond Birth (task 1043)
- [Challenge] Exercise 5-7 Fibonacci Number (task 1030)
- [Challenge] Exercise 5-9 Sine (task 1045)

UT #2 仍然沒有 "full mark" 私人特別獎賞, 不要輕易給 Ms LU 打敗 ... 請你加油

Just do it!

All in one

分組之後的新一輪 on line judge 題目:

- [Challenge] Exercise 4-6 Prime Factors (task 1011)
- [Classwork] Modify EG7-04.pas to add CalculateTax() (task 1073)
- [Homework] Exercise 6-2 Bank Payment Calculation (task 1028)

繼續做「數學化」的2-D array 題目:

- (1) 完成 ex8-1 pascal triangle (task 1085) 之後可以繼續做:
 - task 1098 Square (format tester) (2-D array)
 - task 1009 Magic Square (2-D array) (如果你沒有頭緒, 可以參考我的元謎)

Arrays manipulation

Just do it!

String processing

由於今天有兩位4的同學公假, 請大家今天晚上自行完成第9課 string processing 功課:

- [Classwork] Exercise 9-1 Date Format
- [Classwork] Exercise 9-2 Initial of Name <-
- [Homework] Exercise 9-3 Password <- 輸入20個字符計算密碼
- [Homework] Exercise 9-4 Grade Point <- 會考計分
- [Homework] Exercise 9-5 Roman Numerals (task 1067)

[Judge] task 1071 Serial numbers
[Judge] task 1086 Creating email accounts
[Judge] task 1100 Spy (Episode 1)

online judge ch 10 的相關題目:

- task 1103 We love BUBBLE sort! <- Easy job!
- task 1014 Medal <- multiple keys sorting
- task 1102 Sorting game <- bubble sort is NOT fast enough to solve the last 5 sets of data (TLE error), you must use sorting algorithms of O(n log n) to solve this problem, such as QUICKSORT or HEAPSORT!

Sorting

- ## Useful open resources from the Pui Ching Team
- Pascal learning web & Program tracing : <http://resource.puiching.edu.hk/computer/>
 - Computer discussion forum : <http://student.puiching.edu.hk/viewforum.php?forum=3>
 - [cit] 08-09 中四電腦課 重要資料
 - [alcs] ALCS 2009 Chungsir Class

Demonstration Time ~ Thank you!