**OOP in C++ : အပိုင္း (၂၆) - Functions (4)**

Function ေတြဟာ C++ ရဲ႕ အေျခခံက်တဲ႔ အစိတ္အပိုင္းေတြ ျဖစ္တဲ႔ အတြက္ အေသးစိတ္ သိရွိထားဖို႔လည္း လိုအပ္ပါတယ္။ ဒါေၾကာင္႔ ဒီအပတ္မွာလည္း function ေတြ အေၾကာင္းကိုပဲ ဆက္လက္ေဆြးေႏြးသြားမွာ ျဖစ္ပါတယ္။

**Passing a circle Structure**

အခုတစ္ပတ္မွာေတာ႔ Console Graphics Lite functions မ်ားကို အသံုးျပဳၿပီး ေရးသားထားတဲ႔ ဥပမာေလးကို ေလ႔လာသြားမွာ ျဖစ္ပါတယ္။ Console Graphics Lite functions ရဲ႕ ကုဒ္မ်ားကို ဂရပ္ဖစ္ပိုင္းဆိုင္ရာ နမူနာ ပရိုဂရမ္မ်ားမွာ ထည္႔သြင္းအသံုးျပဳႏိုင္ေအာင္ ဆိုတဲ႔ ရည္ရြယ္ခ်က္နဲ႔ Robert Lafore ေရးသားတဲ႔ Object-Oriented Programming in C++(4th edition) ရဲ႕ Appendix E မွာ ထည္႔သြင္းေပးထားတာပဲ ျဖစ္ပါတယ္။ ေနာက္ပိုင္းမွာေတာ႔ အဲဒီ function မ်ားကို အေသးစိတ္ ရွင္းလင္းေပးမွာ ျဖစ္ေပမယ္႔ အခုအခန္းမွာေတာ႔ function ေတြနဲ႔ ပက္သက္ၿပီး အဓိကထားေလ႔လာေနတာ ျဖစ္တဲ႔အတြက္ ဂရပ္ဖစ္အပိုင္းေရးသားထားခ်က္မ်ားကို အနည္းငယ္ပဲ ေဆြးေႏြးသြားမွာပါ။ အဲဒီ Console Graphics Lite functions ေတြကို အသံုးျပဳဖို႔ အတြက္ေတာ႔ အသံုးျပဳတဲ႔ compiler အေပၚမူတည္ၿပီး msoftcon.h ဒါမွမဟုတ္ borlancon.h တစ္ခုခုကို include ျပဳလုပ္ေပးဖို႔ လိုအပ္ပါတယ္။ ေအာက္ပါ နမူနာ ပရိုဂရမ္ေလးမွာေတာ႔ function တစ္ခုထဲကို structure တစ္ခု ထည္႔သြင္း အသံုးျပဳပံုကို Console Graphics Lite functions ေတြကို အသံုးျပဳထားၿပီး ေရးသားျပထားပါတယ္။ အဲဒီ နမူနာထဲမွာ စက္ဝိုင္းတစ္ခု ကို ကိုယ္စားျပဳတဲ႔ circle structure တစ္ခုကို အသံုးျပဳထားပါတယ္။ စက္ဝိုင္းတစ္ခုကို ေဖာ္ျပႏိုင္ဖို႔ အလယ္မွတ္နဲ႔ radius ေတြကို ထည္႔သြင္းေပးရမွာပါ။ ဒါ႔အျပင္ စက္ဝိုင္းအတြက္ အေရာင္နဲ႔ fill pattern ေတြကိုလည္း ထည္႔သြင္းထားႏိုင္ပါတယ္။ ေအာက္ပါ circstrc.cpp ကို ေလ႔လာၾကည္႔ႏိုင္ပါတယ္။

// circstrc.cpp

// circles as graphics objects

#include “msoftcon.h” // for graphics functions

////////////////////////////////////////////////////////////////

struct circle //graphics circle

{

int xCo, yCo; //coordinates of center

int radius;

color fillcolor; //color

fstyle fillstyle; //fill pattern

};

////////////////////////////////////////////////////////////////

void circ\_draw(circle c)

{

set\_color(c.fillcolor); //set color

set\_fill\_style(c.fillstyle); //set fill pattern

draw\_circle(c.xCo, c.yCo, c.radius); //draw solid circle

}

//--------------------------------------------------------------

int main()

{

init\_graphics(); //initialize graphics system

//create circles

circle c1 = { 15, 7, 5, cBLUE, X\_FILL };

circle c2 = { 41, 12, 7, cRED, O\_FILL };

circle c3 = { 65, 18, 4, cGREEN, MEDIUM\_FILL };

circ\_draw(c1); //draw circles

circ\_draw(c2);

circ\_draw(c3);

set\_cursor\_pos(1, 25); //cursor to lower left corner

return 0;

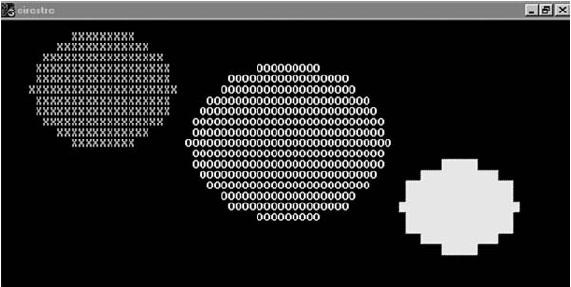
}

အထက္ပါ ဥပမာမွာ circle structure type အမ်ိဳးအစား variables c1, c2 နဲ႔ c3 ေတြကို မတူညီတဲ႔ တန္ဖိုးေတြ ထည္႔သြင္းၿပီး initialized လုပ္ထားပါတယ္။ ဥပမာအေနနဲ႔ c1 ကို ေအာက္ပါအတိုင္း ဖန္တီးလိုက္ပါတယ္-

circle c1 = { 15, 7, 5, cBLUE, X\_FILL };

ကၽြန္္ေတာ္တို႔အေနနဲ႔ console screen ကို 80 columns နဲ႔ 25 rows ရွိတယ္လို႔ သတ္မွတ္လိုက္ပါတယ္။ အထက္က 15 ဟာ column number ဒါမွမဟုတ္ x coordinate ျဖစ္ၿပီး 7 ကေတာ႔ row number ဒါမွမဟုတ္ screen ရဲ႕ ထိပ္ဆံုးက ေနအစျပဳတဲ႔ y coordinate ျဖစ္ပါတယ္။ အဲဒီ တန္ဖိုးႏွစ္ခုက စက္ဝိုင္းရဲ႕ အလယ္မွတ္ကို ေဖာ္ျပေပးတာျဖစ္ပါတယ္။ ေနာက္က 5 ကေတာ႔ စက္ဝိုင္းရဲ႕ အခ်င္းဝက္ တန္ဖိုး ျဖစ္ပါတယ္။ cBLUE ကေတာ႔ စက္ဝိုင္းရဲ႕ အေရာင္ကို ကိုယ္စားျပဳထားၿပီး X\_FILL ကေတာ႔ စက္ဝိုင္းကို ျဖည္႔ေပးမယ္႔ fill pattern ပဲ ျဖစ္ပါတယ္။ စက္ဝိုင္းႏွစ္ခုကို တန္ဖိုး အတူတူေပးၿပီး ဖန္တီးလိုက္ပါတယ္။

စက္ဝိုင္းေတြ အားလံုးကို တန္ဖိုးေတြ သတ္မွတ္ေပးၿပီးတာနဲ႔ circ\_draw() function ကို သံုးႀကိမ္ေခၚယူၿပီး စက္ဝိုင္းေတြကို Console မွာ ဆြဲေပးမွာ ျဖစ္ပါတယ္။ ေအာက္က ပံုေလးမွာေတာ႔ အဲဒီ result ေတြကို ျပသထားပါတယ္။ ပံုေတြက ခပ္ၾကမ္းၾကမ္းျဖစ္ေနေပမယ္႔ အေျခခံ ဂရပ္ဖစ္ နမူနာေလး အျဖစ္ေတာ႔ အသံုးက်ပါတယ္။



ဒီနမူနာေလးမွာ စက္ဝိုင္းတစ္ခုရဲ႕ သြင္ျပင္လကၡဏာေတြကို structure ထဲမွာ စုစည္းထားတာကို သတိထားမိမွာပါ။ ၄င္းကို circ\_draw() function ထဲကို input အေနနဲ႔ ထည္႔သြင္းေပးလိုက္ၿပီး ေရးဆြဲလိုက္တာပဲ ျဖစ္ပါတယ္။ circ\_draw() ကို ဘယ္လို ေရးသားထားသလဲ ဆိုတာကိုေတာ႔ function အေၾကာင္း ၿပီးတဲ႔ အခ်ိန္မွာ Console Graphics Lite functions မ်ားအေၾကာင္း အေသးစိတ္ ရွင္းျပတဲ႔ အထဲမွာ ထည္႔သြင္းရွင္းျပေပးသြားမွာပါ။ ဒီသင္ခန္းစာေလးကိုေတာ႔ ဒီေနရာမွာပဲ ရပ္နားပါရေစ။

ေနာက္အပတ္ေတြမွာလဲ function အေၾကာင္းကိုပဲ ဆက္လက္ ေဆြးေႏြးသြားပါမယ္ခင္ဗ်ာ။ ဤအခန္းဆက္ ေဆာင္းပါးမ်ားဟာ IT ေက်ာင္းသားမ်ားအတြက္ အထူးရည္ရြယ္ၿပီး ေရးသားထားတာ ျဖစ္ပါတယ္။ သင္ရိုးကို ေရးသားျခင္း ျဖစ္တဲ႔အတြက္ တိုက္ရိုက္ ဘာသာျပန္ဆိုျခင္းနည္းကို ေနရာေတာ္ေတာ္မ်ားမ်ားမွာ သံုးထားပါတယ္။ သိလိုသည္မ်ားကို aungwh2013@gmail.com ကို ဆက္သြယ္ၿပီး ေမးျမန္းႏိုင္သလို YCC ေက်ာင္းသားမ်ား အတြက္လည္း ေအာက္ေဖာ္ျပပါ လိပ္စာရွိ ေတာ္ဝင္ ကြန္ပ်ဴတာ စင္တာ၌လည္း ေလ႔လာ စံုစမ္းႏိုင္ပါတယ္ခင္ဗ်ာ။

**အကိုးအကား**

* Object-Oriented Programming in C++(4th edition), Robert Lafore, Copyright©2002 by Sams Publishing: ISBN 0-672-32308-7

Dr. ေအာင္ဝင္းထြဋ္ (bluephoenix)

http://engineer4myanmar.blogspot.com

ေတာ္ဝင္ ကြန္ပ်ဴတာ စင္တာ

၁၇၉ စ၊ သုမဂၤလာ၊ ေစ်းေလး အေနာက္ဘက္၊ ျပင္ဦးလြင္ၿမိဳ႕